AÑO IV - NUM. 111

SEMANAL
135 Ptas.

MICRO HOBBY

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR Y COMPATIBLES

AÑO IV - N.º 111



NUEVO:

GAUNTLET: DESAFÍO EN EL CASTILLO DE LAS SOMBRAS

MICROFILE

BACKUP PARA EL INTERFACE BETA

INICIACIÓN

CÓMO ES LA MEMORIA PAGINADA DEL 128 K

LENGUAJES:

FUNCIONES Y PROCEDIMIENTOS EN PASCAL

JUSTICIEROS DEL SOFTWARE

PROGRAMA DEL AÑO: MOVIE

ZAFI CHIP

¡¡COMANDO SALVAJE!!



—Spectrum Commodore —
Amstrad
Amstrad Disk

Ikari Warriors añade otra dimension: Dos jugadores entran en accion simultaneamente.

ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141, 28046 Madrid
Tel. 459 30 04. Tel. Barna. 209 33 65. Télex: 22690 ZAFIR E

AÑO IV
N.º 111
Del 13 al 19
de enero
de 1987

MICRO HOBBY

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR Y COMPATIBLES

Canarias, Ceuta y Melilla:
130 ptas. Sobretasa aérea para Canarias: 10 ptas.





MICROHOBBY NUMEROS ATRASADOS

Queremos poner en conocimiento de nuestros lectores que para conseguir números atrasados de MICROHOBBY SEMANAL, no tienen más que escribirnos indicándonos en sus cartas el número deseado y la forma de pago elegida de entre las tres modalidades que explicamos a continuación. Una vez tramitado esto, recibirá en su casa el número solicitado al precio de 95 ptas. + 6% de IVA hasta el n.º 36, a 126 ptas. + 8% de IVA hasta el n.º 60 y a 135 ptas. desde el n.º 60 en adelante.

FORMAS DE PAGO

- Enviando talón bancario nominativo a Hobby Press, S. A. al apartado de Correos 54062 de Madrid.
- Mediante Giro Postal, indicando número y fecha del mismo.
- Con Tarjeta de Crédito (VISA o MASTER CHARGE), haciendo constar su número y fecha de caducidad.



Director Editorial: José I. Gómez-Centurión. **Director Ejecutivo:** Domingo Gómez. **Asesor Editorial:** Gabriel Nieto. **Diseño:** Jaime González. **Redacción:** Amalio Gómez, Pedro Pérez, Jesús Alonso. **Secretaria Redacción:** Carmen Santamaría. **Colaboradores:** Primitivo de Francisco, Rafael Prades, Miguel Sepúlveda, Sergio Martínez, J. M. Lazo, Paco Martín. **Corresponsal en Londres:** Alan Heap. **Fotografía:** Carlos Candel, Chema Sacristán. **Portada:** José María Ponce. **Dibujos:** Teo Mójica, F. L. Frontán, J. M. López Moreno, J. Igual, J. A. Calvo, Lóriga, J. Olivares. **Edita:** HOBBY PRESS, S. A. **Presidente:** María Andrino. **Consejero Delegado:** José I. Gómez-Centurión. **Jefe de Producción:** Carlos Peropadre. **Publicidad:** Mar Lumbreras. **Secretaria de Dirección:** Pilar Aristizábal. **Suscripciones:** M.ª Rosa González, M.ª del Mar Calzada. **Redacción, Administración y Publicidad:** Ctra. de Irún Km 12.400, 28049 Madrid. Tél: 734 70 12. Télex: 49480 HOPR. **Pedidos y Suscripciones:** Tél: 734 65 00. **Dto. Circulación:** Paulino Blanco. **Distribución:** Coedis, S. A. Valencia, 245. Barcelona. **Imprime:** Rotedic, S. A. Ctra. de Irún, km 12.450 (MADRID). **Fotocomposición:** Novocomp, S.A. Nicolás Morales, 38-40. **Fotomecánica:** Grof, Ezequiel Solana, 16. Depósito Legal: M-36 598-1984. Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64. 1209 BUENOS AIRES (Argentina). MICROHOBBY no se hace necesariamente solidaria de las opiniones vertidas por sus colaboradores en los artículos firmados. Solicitado control OJD

TASWORD 128: PROCESADOR DE TEXTOS PARA EL SPECTRUM 128



Tasman, la compañía británica más especializada en tratamiento y procesadores de texto, ha creado un nuevo programa para el Spectrum 128 K: el Tasword 128.

Tasman software ha sido la creadora de la gran mayoría de los programas de tratamiento de textos existentes en la actualidad para los diferentes ordenadores de la casa Sinclair: Tasword One, Two y Three para Spectrum o Tascopy para el QL.

Con estos antecedentes, se puede adivinar la gran calidad del nuevo Tasword 128.

Tasword 128 ha sido especialmente desarrollado para utilizar la memoria adicional de 128. De esta forma se consigue crear textos de incluso 60 K, lo que supone que permite escribir hasta 10.000 palabras.

Tasword 128 posee todas las características que han hecho famoso al Tasword Three: comandos sencillos y asequibles, diferentes posibilidades de carga e impresión, justificación de bloques, tabulaciones, inserción de párrafos y todos los elementos que hacen de un procesador de textos una herramienta útil y práctica.

El soporte en el que se suministra este Tasword es una cinta de cassette, aunque permite una rápida y cómoda transferencia a microdrive. En la actualidad su precio en Gran Bretaña es de 20 libras.



ENSAMBLADOR DE CÓDIGO MÁQUINA

Nos encontramos ante una nueva utilidad que deja atrasadas a las anteriores en su campo. Hablamos del Láser Genius de Ocean, un fabuloso ensamblador monitor de Código Máquina que hará las delicias de los usuarios avanzados. Decimos de los usuarios avanzados porque la complicación del programa es mucha a simple vista, aunque con algunas horas delante de la pantalla del ordenador y la ayuda del completísimo manual que se adjunta con el programa (más de 140 hojas), podrás resolver muchas de las dudas planteadas al principio. El paquete consta de un programa cargador-relocalizador, archivos binarios para ser utilizados con el ensamblador, un «toolkit» bastante completo, el propio ensamblador y algunos ejemplos.

De momento, este paquete no está disponible en España, pero esperemos que alguno de los distribuidores españoles haga las negociaciones oportunas para importarlo y traducir las instrucciones, con lo cual harían un gran favor a los viciosos del Código Máquina, que en este país son muchos.

Por último os diremos que el precio del paquete en Inglaterra es de 15 libras, con lo que en España podría costar unas 3.500 ptas. No obstante, la calidad del programa vale ese precio.

MARBLE MADNESS: LA ULTIMA «LOCURA» DE MELBOURNE



Melbourne House, la compañía responsable de éxitos tan recientes como Asterix o The Way of the Exploding Fist II, vuelve al mercado con un nuevo arcade de habilidad.

Se trata de Marble Madness, juego éste que podría ser considerado como la segunda parte del legendario Gyroscope, programa pionero del estilo tan de moda en la actualidad, de juegos basados en laberintos tridimensionales.

Este nuevo arcade posee similares características a las de su predecesor, pues también se desarrolla en un complicado y desnivelado escenario por el cual deberemos desplazar nuestro giratorio elemento protagonista.

Sin embargo, la gran particularidad que presenta este juego es que nos ofrece la posibilidad de crear nuestros propios laberintos; es decir, podemos diseñar un juego a nuestro gusto.

Evidentemente, no tenemos una libertad total para la realización de dichos gráficos, pero podemos elegir entre un considerable número de elementos predefinidos que nos permitirán variar notablemente el circuito inicial.

El desarrollo en sí del juego es similar al de otros arcades de este tipo, pero esta circunstancia le imprime un caracter muy especial.

Uno de los programas a tener en cuenta para este año recién iniciado.

FORMULA 2 JOYSTICK: LA NOVEDAD DE KEMPSTON



Kempston Micro Electronics, una de las compañías más importantes en cuanto a lo que al diseño de periféricos se refiere, acaba de presentar en el mercado un nuevo modelo de joystick que viene a sumarse a su ya nutrida gama.

La nueva creación ha sido bautizada con el nombre de Fórmula 2, el cual, como su propio nombre indica, está especialmente diseñado para su utilización en programas de acción rápida, tales como simuladores o arcades.

Su diseño ergonómico le imprime una buena adaptabilidad a la mano y su botón de disparo, situado en la parte superior del mismo, hace muy cómodo su manejo. Sin embargo, Fórmula 2 posee otros dos disparadores situados en la base del joystick, colocados al efecto de que resulten asequibles tanto para los diestros como para los zurdos.

Sus características principales son las siguientes:

— Ocho micrositches que permiten otras tantas direcciones de movimiento.

— Firme sujeción a las superficies planas gracias a cuatro potentes ventosas.

— Muelle de retorno para que el mango recobre su posición central.

— Compatibilidad con Commodore, Atari y Sinclair Spectrum cuando se utiliza el correspondiente interface de joystick Kempston.

Aquí LONDRES

«Star Soldier» es la última publicación de Arguss Press Software bajo la compañía de Quicksilva. Se trata de un juego típicamente arcade formado por 99 escenarios diferentes. Se puede conseguir para el Commodore-64 a un precio de 9 libras.

El «boom» de ordenadores ha empezado de nuevo. La demanda de estas máquinas ha aumentado considerablemente, si se compara con la del año pasado. Por otra parte, este aumento supone una baja importante para el Spectrum Plus II. Los distribuidores nacionales encuentran dificultades a la hora de abastecer las demandas. El problema parece acentuarse con el Amstrad, por lo que los distribuidores no tendrán que hacer otra cosa más que esperar a ver cómo se desarrollan los acontecimientos y tendencias del mercado.

Micro-Prose, la compañía de software famosa por sus productos de simulación, ha roto las relaciones con U.S. Gold. Esta ha estado vendiendo títulos de Micro-Prose durante dos años en el Reino Unido, pero desde primeros de año se hará independiente. La separación ha sido enteramente amistosa, ya que se ha llegado a la conclusión de que Micro-Prose puede dedicar mayor soporte y tiempo a sus propios productos.

***«Planit» es un nuevo* conjunto de programas para los ordenadores Amstrad PCW y CPC, diseñado para organizar la vida cotidiana y que consiste en una agenda, un índice alfabético y una hoja de cálculo. La hoja de cálculo puede contener hasta 24 cuentas individuales, la agenda acoge 15 entradas diarias y el índice alfabético posee un libro de direcciones, una lista de teléfonos y permite la impresión de direcciones. «Planit» es una publicación de Database.**

Alan Heap

MICROPANORAMA

SCOOBY DOO EN EL CASTILLO MISTERIOSO

Hanna-Barbera es, sin duda, una de las más antiguas y prestigiosas compañías mundiales dedicadas a la creación de dibujos animados. Uno de sus personajes más conocidos y entrañables es Scooby Doo. Quién no recuerda el famoso grito: Scooby Doo, ¿dónde estáaas?

Pues ahora, a través de la no menos popular casa Elite, vamos a tener la posibilidad de disfrutar en nuestro Spectrum con las divertidas aventuras de este enorme y miedoso can.

El juego, con estructura de arcade, nos traslada a un tétrico y sombrío castillo plagado de fantasmas, monstruos y demás seres metedores de miedo.

Nuestra misión consistirá en eludir a todos estos individuos para poder recorrer todas las plantas del castillo en busca de el resto de los componentes de la pandilla.

Los diseños de Scooby Doo son prácticamente idénticos a los de los dibujos animados, así como los movimientos del mismo, los cuales reflejan a la perfección las piruetas y forma de correr de este simpático personaje.

Este programa ha sido realizado para Spectrum, Amstrad y Commodore, aunque, por el momento, no está disponible en nuestro país.



LOS 20 +

CLASIFICACIÓN	SEMANAS PERM.	TENDENCIA	LOS 20 +	SPECTRUM	AMSTRAD	COMMODORE	MSX
1	7	–	**SUPER-10.** Erbe	●			
2	11	↑	**DRAGON'S LAIR.** Software Proyects	●		●	
3	20	↓	**T.S.A.M. II.** U.S. Gold	●			
4	24	–	**GREEN BERET.** Imagine	●			
5	9	↑	**KNIGHT RIDER.** Ocean	●			
6	4	↑	**INFILTRATOR.** U.S. Gold	●			
7	1	↑	**T.S.A.M. III.** U.S. Gold	●			
8	7	↓	**ASTERIX.** Melbourne House	●			
9	7	↓	**ANTIRIAD.** Palace Software	●	●		
10	7	↓	**NIGHTMARE RALLY.** Ocean	●	●		
11	8	↑	**GHOSTS'N GOBLINS.** Elite	●	●	●	
12	8	↓	**STREET HAWK.** Ocean	●	●		
13	11	↓	**TENNIS.** Imagine	●		●	
14	4	↓	**GREAT ESCAPE.** Ocean	●			
15	1	↑	**COBRA.** Ocean	●			
16	4	↓	**URIDIUM.** Hewson	●		●	
17	1	↑	**BREAKTHRU.** U.S. Gold	●	●		
18	3	↓	**FIRELORD.** Hewson	●	●	●	
19	1	↑	**GOONIES.** U.S. Gold	●	●	●	
20	1	↑	**4 SUPER 4.** Dinamic	●			

Esta información ha sido elaborada con la colaboración de los centros de Microinformática de El Corte Inglés.

El Corte Inglés



ALIENS
EL REGRESO
DEL GRAN EXITO EN EL CINE,
AHORA MAS CERCA: EN TU ORDENADOR
¡ES TU GUERRA!
Disponibles para:
COMMODORE C
SPECTRUM S
AMSTRAD A
EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES, O DIRECTAMENTE POR CORREO O TELEFONO A: PROEIN, S.A.
Distribuido en Cataluña por: DISCOVERY INFORMATIC C/. Arco Iris, 75 - BARCELONA - Tels. 256 49 08 / 09
Velázquez, 10 - 28001 Madrid - Tels. (91) 276 22 08 / 09

SATURNO 2

David CALANDRA

En el planeta Saturno, hace ya muchos cientos de años, se excavaron unas minas en su suelo, con el fin de buscar los preciosos minerales que éste escondía en sus entrañas.

Para esta difícil misión la NADA ha enviado a su famoso explorador espacial Manolo Minglanillas, osado tripulante que se ha enfrentado en multitud de ocasiones a los extraterrestres de todas las galaxias conocidas.

El juego consiste en llevar a Manolo por los pasadizos secretos de las 42 pantallas, en busca de los 5 diamantes que se encuentran esparcidos por sus rincones. Una vez recogidos los diamantes nos dirigiremos de nuevo al cohete y nos prepararemos para un final feliz.

Para que el juego no se limite simplemente a recoger diamantes, también tenemos que evitar ser atrapados por los saturninos salvajes que se esconden en las grietas del planeta, además de que tenemos la obligación de cumplir la misión antes de que se nos agote el oxígeno.

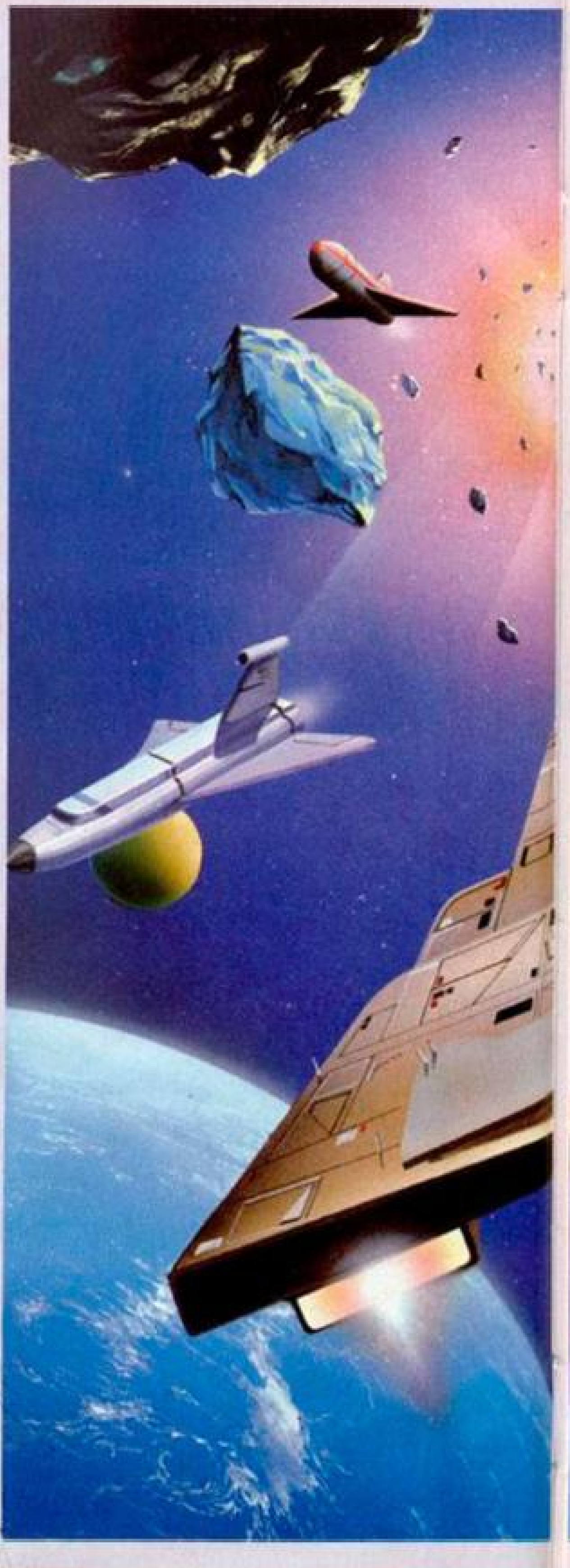
Las teclas de movimiento son redefinibles.

Las líneas de listado de cargador universal que no estén en el listado deben teclearse introduciendo como datos 200 y 0 para control.

Todas las letras mayúsculas subrayadas deben teclearse en modo gráfico.

LISTADO 1

```
  10 REM © 1985 DAVID  CALANDRA
REULA - MINGLANILLLAS EN SATURNO
  15 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLEAR 64599: LOAD ""COD
E 64800,487
  20 LET d$="U": LET i$="O": LET
 s$="0": LET RE=0: LET b$=":;<=:
;:;>:;<>:;=>:;:;=:;>=:;<>:;=:;=>
:;<:;:;=<:;=>:;<:;:;<=:;>:;": LE
T a$="$$$$$$$$$$$$$$"
  25 POKE 23609,0: POKE 23617,23
0: POKE 23658,8: GO SUB 8950
  30 BEEP .1,0: CLS : PRINT AT 5
,4; INK 4;"MINGLANILLAS EN SATUR
NO";AT 8,2; INK 6;"© 1985  DAVID
 CALANDRA REULA";AT 11,6; INK 5;
"PUNTUACION MAXIMA "; INK 4;RE;A
T 12,2; INK 5;"CONSEGUIDA POR ";
 INK 3;A$;AT 15,4; INK 4;"D ....
.. DEFINIR TECLADO";AT 17,7; INK
 6;"PULSA J PARA JUGAR"
  35 IF INKEY$="J" THEN GO TO 90
  40 IF INKEY$="D" THEN GO TO 50
  45 GO TO 35
  50 CLS : PRINT AT 10,9; INK 6;
"DEFINIR TECLADO": INK 5: INPUT
"TECLA IZQUIERDA "; LINE i$: INP
UT INK 6;"TECLA DERECHA "; LINE
d$: INPUT INK 4;"TECLA SALTO ";
LINE s$: GO TO 25
  90 FOR A=0 TO 5: BEEP .01,5: B
EEP .01,30: NEXT A
 103 LET k1=1: LET k2=1: LET k3=
1: LET k4=1: LET k5=1: LET hir=0
: LET air=8: LET qq=1: LET pin=0
: LET go=0: LET com=0: LET dad=0
: LET x1=13: LET y1=8: LET vi=3:
 LET pu=0: LET di=0: LET ai=184
 104 LET tas=130: LET cal=9040:
LET lin=9040: LET z1=7010: LET z
2=7020: LET z3=7040: LET z4=7060
: LET z5=7070: LET z6=7080: LET
z7=7090: LET z8=7100
 105 CLS : PRINT AT 19,0; INK 4;
"PUNTOS";AT 19,7; INK 6;"MAXIMO"
;AT 19,14; INK 5;"DIAMANTES";AT
20,2; INK 6;pu;AT 20,9; INK 5;re
;AT 20,18; INK 4;di;AT 21,0; INK
 3;"AIRE";AT 21,5; INK 5;"[redacted]
[redacted]": PRINT #0; INK 4;"
 © 1985  DAVID CALANDRA REULA"
 110 INK 6: GO SUB 8900: PRINT A
T 19,24;"_£ _£ _£";AT 20,24;"ab
ab ab";AT 21,24;"cd cd cd";AT 18
,0; INK 4;"IIIIIIIIIIIIIIIIIIIII
IIIIIIIIIII"
 120 GO SUB 130: GO TO 6000
```






```
 135 LET z9=7120: LET l8=1: GO S
UB 8989: PRINT AT 16,0; INK 3;b$
 140 LET mu=0: LET to=1: LET ag=
0: LET tol=22: LET ra=13: LET zo
=26: LET par=9010: INK 4: PLOT 3
1,160: DRAW 0,15: DRAW 1,0: DRAW
 0,-15: PRINT AT 2,3;"UV"
 145 PRINT AT 3,3;"UX";AT 4,3;"Y
Z";AT 5,2;"[redacted]";AT 6,2;"[redacted]";A
T 7,3;"[redacted]";AT 8,3;"[redacted]";AT 9,3;"[redacted]";AT 10,3;"[redacted]";AT 11,2;"[redacted]";A
T 12,2;"[redacted]";AT 13,2;"[redacted]";AT
14,2;"[redacted]";AT 15,2;"[redacted]"
 150 GO SUB 9600: INK 4: PLOT IN
VERSE 1;24,150: DRAW INVERSE 1;0
,-100: PLOT INVERSE 1;39,150: DR
AW INVERSE 1;0,-100: PLOT INVERS
E 1;36,138: DRAW INVERSE 1;0,-80
 170 GO SUB 9950: RETURN
 505 GO SUB 8950: BEEP .5,-5: CL
S : PRINT AT 7,3; INK 4;"ENHORAB
UENA, HAS CONSEGUIDO";AT 9,9;"LO
S 5 DIAMANTES";AT 13,9; INK 6;"P
ULSA UNA TECLA" : PAUSE 0
 510 CLS : FOR y=0 TO 30: BEEP
```



```
002,50: PRINT AT RND*21,RND*31;
INK (RND*6)+1;".": NEXT y: PRINT
 AT 10,9; INK 5;"BUEN VIAJE !!!"
: FOR y=0 TO 60 STEP 2.5: BEEP .
05,y: NEXT y: PAUSE 0: LET vi=-1
1005 FOR a=30 TO 20 STEP -1: BEE
P .01,a: BEEP .01,a+10: NEXT a
1010 LET vi=vi-1: IF vi=2 THEN P
RINT AT 19,24;"  ";AT 20,24;"  "
;AT 21,24;"  ": GO TO 1070
1030 IF vi=1 THEN PRINT AT 19,27
;"  ";AT 20,27;"  ";AT 21,27;"
": GO TO 1070
1040 IF vi=0 THEN PRINT AT 19,30
;"  ";AT 20,30;"  ";AT 21,30;"
": GO TO 1070
1055 IF pu<=re THEN GO TO 25
1060 GO SUB 8950: LET re=pu: CLS
 : PRINT AT 8,2; INK 6;"ENHORABU
ENA, HAS CONSEGUIDO           UNA
 ALTA PUNTUACION": PRINT : PRINT
 INK 5;"      INTRODUCE TU NOMB
RE": INPUT LINE a$: GO TO 25
1075 GO SUB 6990: IF ai<41 THEN
PRINT INK 5;AT 21,5;"[redacted]": LET ai=184
```

```
1080 IF l8=1 THEN LET x1=13: LET
 y1=28: LET qq=0: GO SUB 9950: G
O TO 6800
1090 IF l8=0 THEN LET x1=13: LET
 y1=2: LET qq=1: GO SUB 9950: GO
 TO 6800
2020 IF (ATTR (x1+3,y1)=69 OR AT
TR (x1+3,y1)=70) AND (ATTR (x1+3
,y1+1)=69 OR ATTR (x1+3,y1+1)=70
) THEN LET hir=0: PRINT AT x1,y1
; INK 6;"  ": LET x1=x1+1: GO SU
B 9950: GO SUB 5000: GO TO 2010
2025 GO TO 6800
3010 IF ATTR (x1-1,y1)=66 OR ATT
R (x1-1,y1+1)=66 OR ATTR (x1-1,y
1)=68 OR ATTR (x1-1,y1+1)=68 THE
N GO TO 2000
3030 IF ATTR (x1-1,y1+2)=66 OR A
TTR (x1-1,y1+2)=68 OR ATTR (x1,y
1+2)=66 OR ATTR (x1+1,y1+2)=66 O
R ATTR (x1,y1+2)=68 OR ATTR (x1+
1,y1+2)=68 THEN GO TO 2000
3040 GO TO san
3500 IF ATTR (x1-1,y1)=66 OR ATT
R (x1-1,y1+1)=66 OR ATTR (x1-1,y
1)=68 OR ATTR (x1-1,y1+1)=68 THE
N GO TO 2000
3510 IF ATTR (x1-1,y1-1)=66 OR A
TTR (x1-1,y1-1)=68 OR ATTR (x1,y
1-1)=66 OR ATTR (x1+1,y1-1)=66 O
R ATTR (x1,y1-1)=68 OR ATTR (x1+
1,y1-1)=68 THEN GO TO 2000
3520 GO TO san
3900 IF ATTR (x1+3,y1)=67 OR ATT
R (x1+3,y1+1)=67 OR ATTR (x1+3,y
1)=68 OR ATTR (x1+3,y1+1)=68 THE
N LET hir=0: GO TO 6000
3910 GO TO san
4002 LET hir=1: IF qq=0 THEN GO
TO 4500
4005 LET san=4010: GO TO 3000
4020 GO SUB 6990: LET x1=x1-1: L
ET y1=y1+1: GO SUB 9500
4027 LET san=4030: GO TO 3000
4030 GO SUB 6990: LET x1=x1-1: G
O SUB 9500
4037 LET san=4040: GO TO 3000
4040 GO SUB 6990: LET x1=x1-1: L
ET y1=y1+1: GO SUB 9500
4047 LET san=4050: GO TO 3000
4050 GO SUB 6990: LET x1=x1-1: G
O SUB 9500
4057 LET san=4060: GO TO 3000
4060 GO SUB 6990: LET x1=x1-1: L
ET y1=y1+1: GO SUB 9500
4067 LET san=4070: GO TO 3030
4070 GO SUB 6990: LET y1=y1+1: G
O SUB 9500
4076 LET san=4077: GO TO 3900
4077 LET san=4078: GO TO 3030
4078 IF ATTR (x1+3,y1+2)=68 THEN
 GO SUB 6990: LET y1=y1+1: GO SU
B 9950: LET hir=0: GO TO 6800
4080 GO SUB 6990: LET x1=x1+1: L
ET y1=y1+1: GO SUB 9500
4086 LET san=4087: GO TO 3900
4087 LET san=4090: GO TO 3030
4090 GO SUB 6990: LET x1=x1+1: G
O SUB 9500
4096 LET san=4097: GO TO 3900
4097 LET san=4100: GO TO 3030
4100 GO SUB 6990: LET x1=x1+1: L
ET y1=y1+1: GO SUB 9500
4106 LET san=4107: GO TO 3900
4107 LET san=4110: GO TO 3030
4110 GO SUB 6990: LET x1=x1+1: G
O SUB 9500
4116 LET san=4117: GO TO 3900
4117 LET san=4120: GO TO 3030
4120 GO SUB 6990: LET x1=x1+1: L
ET y1=y1+1: GO SUB 9500
4499 LET hir=0: GO TO 6800
4500 LET san=4510: GO TO 3500
4510 GO SUB 6990: LET x1=x1-1: L
ET y1=y1-1: GO SUB 9550
4530 LET san=4540: GO TO 3500
4540 GO SUB 6990: LET x1=x1-1: G
O SUB 9550
4560 LET san=4570: GO TO 3500
4570 GO SUB 6990: LET x1=x1-1: L
ET y1=y1-1: GO SUB 9550
4590 LET san=4600: GO TO 3500
4600 GO SUB 6990: LET x1=x1-1: G
O SUB 9550
4620 LET san=4630: GO TO 3500
4630 GO SUB 6990: LET x1=x1-1: L
ET y1=y1-1: GO SUB 9550
4650 LET san=4660: GO TO 3510
4660 GO SUB 6990: LET y1=y1-1: G
O SUB 9550
4675 LET san=4680: GO TO 3900
4680 LET san=4685: GO TO 3510
4685 IF ATTR (x1+3,y1-1)=68 THEN
 GO SUB 6990: LET y1=y1-1: GO SU
B 9950: LET hir=0: GO TO 6800
4690 GO SUB 6990: LET x1=x1+1: L
ET y1=y1-1: GO SUB 9550
4705 LET san=4710: GO TO 3900
4710 LET san=4720: GO TO 3510
4720 GO SUB 6990: LET x1=x1+1: G
O SUB 9550
4735 LET san=4740: GO TO 3900
4740 LET san=4750: GO TO 3510
4750 GO SUB 6990: LET x1=x1+1: L
ET y1=y1-1: GO SUB 9550
4765 LET san=4770: GO TO 3900
4770 LET san=4780: GO TO 3510
4780 GO SUB 6990: LET x1=x1+1: G
O SUB 9550
4795 LET san=4800: GO TO 3900
4800 LET san=4810: GO TO 3510
4810 GO SUB 6990: LET x1=x1+1: L
ET y1=y1-1: GO SUB 9550
4999 LET hir=0: GO TO 6800
5010 IF mu=0 THEN GO TO 5100
5015 INK 5: IF go=0 THEN GO TO 5
050
5020 PRINT AT va,li;"  ";AT va+1
,li;"  ": LET va=va-1: PRINT AT
va,li;"?@";AT va+1,li;"AB"
5025 IF va=4 THEN GO SUB 9980: L
ET go=0
5040 GO TO 5080
5055 PRINT AT va,li;"  ";AT va+1
,li;"  ": LET va=va+1: PRINT AT
va,li;"?@";AT va+1,li;"AB"
5060 IF va=14 THEN GO SUB 9980:
LET go=1
5080 IF y1+1=li AND (x1=va+1 OR
x1=va OR x1+1=va OR x1+2=va) OR
y1=li AND (x1-1=va OR x1+2=va) O
R y1-1=li AND (x1-1=va OR x1=va
OR x1+1=va OR x1+2=va) THEN LET
hir=0: GO TO 1000
5100 IF to=0 THEN GO TO 5200
5105 IF pin=0 THEN GO TO 5150
5110 PRINT AT 15,tol;"   ": LET
tol=tol-1: PRINT AT 15,tol; INK
4; BRIGHT 0;"QNO": BRIGHT 1
5115 IF tol=ra THEN GO SUB 9980:
 LET pin=0
5145 GO TO 5180
5150 PRINT AT 15,tol;"   ": LET
tol=tol+1: PRINT AT 15,tol; INK
4; BRIGHT 0;"NOP": BRIGHT 1
5160 IF tol=zo THEN GO SUB 9980:
 LET pin=1
5190 IF x1+2=15 AND (y1=tol OR y
1=tol+1 OR y1=tol+2 OR y1+1=tol)
 THEN LET hir=0: GO TO 1000
5195 IF hir=0 AND x1+3=15 AND (y
1=tol OR y1=tol+1 OR y1=tol+2 OR
 y1+1=tol) THEN GO TO 1000
5200 IF ag=0 THEN GO TO 5300
5210 IF ATTR (x1+3,y1)=93 AND AT
TR (x1+3,y1+1)=93 THEN INK 6: GO
 TO 5220
5215 GO TO 5240
5220 IF qq=1 THEN PRINT AT x1,y1
;"  ";AT x1+1,y1;"kl";AT x1+2,y1
;"mn": BEEP .1,5: BEEP .1,6: PRI
NT AT x1+1,y1;"  ";AT x1+2,y1;"k
l": BEEP .1,4: BEEP .1,5: PRINT
AT x1+2,y1;"  ": BEEP .1,3: BEEP
 .1,4: GO TO 1010
5230 IF qq=0 THEN PRINT AT x1,y1
;"  ";AT x1+1,y1;"_£";AT x1+2,y1
;"ab": BEEP .1,5: BEEP .1,6: PRI
NT AT x1+1,y1;"  ";AT x1+2,y1;"_
£": BEEP .1,4: BEEP .1,5: PRINT
AT x1+2,y1;"  ": BEEP .1,3: BEEP
 .1,4: GO TO 1010
5300 IF y1=30 THEN LET y1=1: GO
SUB 8990: LET l8=0: GO SUB par
5310 IF y1=0 THEN LET y1=29: GO
SUB 8990: LET l8=0: GO SUB tas
5320 IF x1=15 THEN LET x1=1: GO
SUB 8990: LET l8=1: GO SUB cal
5330 IF x1=0 THEN LET x1=14: GO
SUB 8990: LET l8=0: GO SUB lin
5400 IF air<10 THEN GO TO 5410
5405 INK 5: LET ai=ai-1: PLOT OV
ER 1;ai,0: DRAW OVER 1;0,7
5410 IF ai<41 THEN GO TO 1000: P
RINT INK 5;AT 21,5;"[redacted]": LET ai=184
5415 LET air=air-1: IF air=0 THE
N LET air=10
5430 IF hir=1 THEN GO TO 5500
5470 IF (ATTR (x1+3,y1)=69 OR AT
TR (x1+3,y1)=70) AND (ATTR (x1+3
,y1+1)=69 OR ATTR (x1+3,y1+1)=70
) THEN PRINT AT x1,y1; INK 6;"
": LET x1=x1+1: GO SUB 9950: GO
TO 5000
5600 IF di=5 AND par=9010 AND y1
<7 THEN GO TO 500
5610 IF lin=9090 AND k1=1 AND x1
>11 AND y1>26 THEN PRINT AT 14,2
8;"  ";AT 15,28;"  ": GO SUB 995
0: GO SUB 9700: LET k1=0
5620 IF par=9210 AND tas=9170 AN
D k2=1 AND x1>11 AND y1>25 AND y
1<30 THEN PRINT AT 14,27;"  ";AT
 15,27;"  ": GO SUB 9950: GO SUB
 9700: LET k2=0
5630 IF tas=9260 AND cal=9250 AN
D k3=1 AND x1>6 AND x1<11 AND y1
>13 AND y1<17 THEN PRINT AT 9,15
;"  ";AT 10,15;"  ": GO SUB 9950
: GO SUB 9700: LET k3=0
5640 IF tas=9340 AND k4=1 AND x1
>6 AND x1<11 AND y1>18 AND y1<22
 THEN PRINT AT 9,20;"  ";AT 10,2
0;"  ": GO SUB 9950: GO SUB 9700
: LET k4=0
5650 IF par=9380 AND k5=1 AND y1
>1 AND y1<5 AND x1>11 THEN PRINT
 AT 14,3;"  ";AT 15,3;"  ": GO S
UB 9950: GO SUB 9700: LET k5=0
5900 RETURN
6200 IF INKEY$=d$ AND (ATTR (x1,
y1+2)=69 OR ATTR (x1,y1+2)=70) T
HEN LET qq=1: GO TO 6220
6210 GO TO 6400
6220 INK 6: PRINT AT x1+2,y1;"  "
: LET y1=y1+1: PRINT AT x1,y1-1;
" kl";AT x1+1,y1-1;" mn"
6230 IF dad=0 THEN PRINT AT x1+2
,y1;"op": LET dad=1: GO TO 6400
6240 IF dad=1 THEN PRINT AT x1+2
,y1;"qr": LET dad=2: GO TO 6400
6250 IF dad=2 THEN PRINT AT x1+2
,y1;"st": LET dad=3: GO TO 6400
6260 PRINT AT x1+2,y1;"uv": LET
dad=0: GO TO 6400
6410 IF INKEY$=s$ THEN GO TO 400
0
6420 IF par=9010 AND di<5 AND y1
=6 THEN GO TO 6700
6450 IF INKEY$=i$ AND (ATTR (x1,
y1-1)=69 OR ATTR (x1,y1-1)=70) T
HEN LET qq=0: GO TO 6470
6460 GO TO 6700
6470 INK 6: LET y1=y1-1: PRINT A
T x1,y1;"_£ ";AT x1+1,y1;"ab "
6480 IF com=0 THEN PRINT AT x1+2
,y1;"cd  ": LET com=1: GO TO 6700
6490 IF com=1 THEN PRINT AT x1+2
```

```
,y1;"ef  ": LET com=2: GO TO 6700
6500 IF com=2 THEN PRINT AT x1+2,y1;"gh  ": LET com=3: GO TO 6700
6510 PRINT AT x1+2,y1;"ij  ": LET com=0: GO TO 6700
6700 GO SUB 5000
6900 GO TO 6000
6990 INK 6: PRINT AT x1,y1;"  ";AT x1+1,y1;"  ";AT x1+2,y1;"  ": RETURN
7010 INK 2: PRINT AT 0,0;"DKFDFKFKDKFFKDFKKDFKDKFDFKDFDKDKDKFDGCDFKG CDKFDKDFGCKDFDKFDFGCDFGHECGE   CKDGECG  EECKFGEH  EE      H                CG": RETURN
7020 INK 2: PRINT AT 0,0;"KDFDKFDKFKDFKDFG";AT 1,0;"KDFKGCKDFDFGECG";AT 2,0;"EHCG  ECGEH";AT 0,25;"CDKFDKF";AT 1,26;"HECDFK";AT 2,29;"CGE": RETURN
7040 INK 2: PRINT AT 0,0;"KDFDKFG";AT 1,0;"GCFGEH";AT 2,2;"H";AT 0,16;"CKDFDKFDFKFDKFDF";AT 1,17;"CGECDKFDGEHECFD";AT 2,21;"HECG    HE": RETURN
7060 INK 4: PRINT AT xx,yy;"C=<><G";AT xx+1,yy+1;"HECG": RETURN
7070 INK 4: PRINT AT xx,yy;"C<=<>=G";AT xx+1,yy+1;"CGHEH": RETURN
7080 INK 2: PRINT AT 1,yy;"KJ";AT 2,yy;"JK";AT 3,yy;"KK";AT 4,yy;"JD";AT 5,yy;"JK";AT 6,yy;"FK";AT 7,yy;"JJ";AT 8,yy;"JK";AT 9,yy;"DJ";AT 10,yy;"KK";AT 11,yy;"JJ";AT 12,yy;"FK";AT 13,yy;"DK";AT 14,yy;"JK";AT 15,yy;"FK"
7085 RETURN
7090 INK 3: PRINT AT 16,0;"<=<>=>><=><=>=<=><===<>=<==<=><>==><=>=<==>=<=>=<=<>==<=<=>=<=>=": RETURN
7100 INK 3: PRINT AT 16,0;"<=>==><=>=<=>=<\";AT 17,0;"=<=>=<=>=>==<=>K";AT 16,25;"[=<=>=<";AT 17,25;"J>=<>=>": RETURN
7120 INK 3: PRINT AT 16,0;"=<>=<>\";AT 17,0;"=>=<=>K";AT 16,16;"[=>=<=>=>>=<=>==";AT 17,16;"J=>=<=>=<=>>=<=>": RETURN
8900 POKE 23606,88: POKE 23607,251: RETURN
8950 POKE 23606,0: POKE 23607,60: RETURN
8990 FOR z=0 TO 17: PRINT AT z,0;"                                ": NEXT z: RETURN
9010 LET par=9030: LET tas=130: LET cal=9020: LET ag=1: LET to=0: LET mu=1: LET va=7: LET li=10: GO SUB z1: LET xx=11: LET yy=18: GO SUB z4: GO SUB z8: PRINT AT 16,4; INK 5; PAPER 3;"L[redacted]": GO SUB 9950: RETURN
9020 LET mu=0: LET to=1: LET tol=8: LET ra=1: LET zo=11: LET ag=1: LET tas=9040: LET lin=9010: GO SUB z2: LET yy=30: GO SUB z6: LET xx=6: LET yy=19: GO SUB z4: LET xx=11: LET yy=12: GO SUB z5: GO SUB z7: PRINT AT 16,16; INK 5; PAPER 3;"L[redacted]": GO SUB 9950: RETURN
9030 LET ag=0: LET to=1: LET mu=0: LET tol=25: LET ra=18: LET zo=27: LET tas=9010: LET cal=9050: GO SUB z1: LET yy=30: GO SUB z6: LET xx=11: LET yy=8: GO SUB z5: LET xx=9: LET yy=21: GO SUB z4: GO SUB z9: GO SUB 9950: RETURN
9040 LET ag=1: LET to=0: LET mu=1: LET va=8: LET li=25: LET par=9020: LET cal=9100: GO SUB z1: LET yy=0: GO SUB z6: LET xx=8: LET yy=6: GO SUB z4: LET xx=11: LET yy=16: GO SUB z5: GO SUB z8: PRINT AT 16,5; INK 5; PAPER 3;"L[redacted]": GO SUB 9950: RETURN
9050 LET ag=0: LET mu=1: LET to=0: LET li=15: LET va=7: LET lin=9030: LET par=9060: LET cal=9070: GO SUB z3: LET yy=0: GO SUB z6: LET xx=6: LET yy=9: GO SUB z4: LET xx=11: LET yy=17: GO SUB z5: GO SUB z8: GO SUB 9950: RETURN
9060 LET tas=9050: LET cal=9090: LET ag=1: LET to=0: LET mu=1: LET va=7: LET li=7: GO SUB z1: LET yy=30: GO SUB z6: LET xx=11: LET yy=17: GO SUB z5: GO SUB z8: PRINT AT 16,5; INK 5; PAPER 3;"L[redacted]": GO SUB 9950: RETURN
9070 GO SUB 9020: LET lin=9050: LET tas=9080: RETURN
9080 LET par=9070: LET tas=9100: LET ag=1: LET to=0: LET mu=1: LET va=7: LET li=15: GO SUB z1: LET xx=11: LET yy=8: GO SUB z5: GO SUB z7: PRINT AT 16,18; INK 5; PAPER 3;"L[redacted]": GO SUB 9950: RETURN
9090 LET ag=0: LET to=1: LET mu=0: LET tol=25: LET ra=17: LET zo=27: LET lin=9060: LET cal=9110: GO SUB z2: LET yy=0: GO SUB z6: LET yy=30: GO SUB z6: GO SUB z9: LET xx=6: LET yy=18: GO SUB z5: LET xx=11: LET yy=9: GO SUB z4: GO SUB 9950: RETURN
9100 LET lin=9040: LET par=9080: LET cal=9120: LET ag=1: LET to=0: LET mu=1: LET va=7: LET li=25: GO SUB z2: LET yy=0: GO SUB z6: GO SUB z9: LET xx=11: LET yy=7: GO SUB z5: LET xx=6: LET yy=18: GO SUB z4: PRINT AT 16,22; INK 5; PAPER 3;"L[redacted]": GO SUB 9950: RETURN
9110 IF k1=0 THEN GO TO 9115
9111 INK 6: PRINT AT 14,28;"RS";AT 15,28;"[redacted]R"
9115 LET lin=9090: LET ag=1: LET to=0: LET mu=1: LET va=10: LET li=25: GO SUB z3: LET yy=0: GO SUB z6: LET yy=30: GO SUB z6: GO SUB z7: LET xx=6: LET yy=8: GO SUB z5: LET xx=11: LET yy=18: GO SUB z4: GO SUB 9950: PRINT AT 16,4; INK 5; PAPER 3;"L[redacted]": RETURN
9120 GO SUB 9050: LET lin=9100: LET par=9130: LET cal=9140: RETURN
9130 LET tas=9120: LET par=9160: LET cal=9170: LET ag=0: LET to=0: LET mu=1: LET va=5: LET li=13: GO SUB z1: GO SUB z8: LET xx=11: LET yy=18: GO SUB z4: LET xx=9: LET yy=5: GO SUB z5: GO SUB 9950: RETURN
9140 GO SUB 9090: LET lin=9120: LET cal=9150: RETURN
9150 GO SUB 9115: LET lin=9140: RETURN
9160 GO SUB 9060: LET tas=9130: LET cal=9180: RETURN
9170 GO SUB 9100: LET lin=9130: LET par=9180: LET cal=9190: RETURN
9180 IF k2=0 THEN GO TO 9185
9181 INK 6: PRINT AT 14,27;"RS";AT 15,27;"[redacted]R"
9185 LET tas=9170: LET lin=9160: LET par=9210: LET ag=1: LET to=1: LET mu=0: LET tol=7: LET ra=3: LET zo=16: GO SUB z2: GO SUB z7: LET xx=6: LET yy=17: GO SUB z5: LET xx=11: LET yy=8: GO SUB z4: PRINT AT 16,20; INK 5; PAPER 3;"L[redacted]": GO SUB 9950: RETURN
9190 GO SUB 9120: LET lin=9170: LET par=9200: LET cal=9240: RETURN
9200 GO SUB 9130: LET tas=9190: LET cal=9250: LET par=9230: RETURN
9210 GO SUB 9130: LET tas=9180: LET par=9220: LET cal=9230: RETURN
9220 GO SUB 9185: LET tas=9210: LET par=9300: LET lin=9310: RETURN
9230 GO SUB 9020: LET lin=9210: LET tas=9200: RETURN
9240 GO SUB 9020: LET lin=9190: LET tas=9270: RETURN
9250 LET lin=9200: LET par=9260: LET to=1: LET ag=0: LET mu=0: LET tol=10: LET ra=5: LET zo=25: LET xx=6: LET yy=19: GO SUB z4: LET xx=11: LET yy=10: GO SUB z5: GO SUB z2: LET yy=0: GO SUB z6: GO SUB z7: GO SUB 9950: RETURN
9260 GO SUB 9080: LET tas=9250: LET par=9280: RETURN
9270 GO SUB 9080: LET tas=9410: LET par=9240: RETURN
9280 IF k3=0 THEN GO TO 9285
9281 INK 6: PRINT AT 9,15;"RS";AT 10,15;"[redacted]R"
9285 GO SUB 9020: LET lin=9290: LET tas=9260: RETURN
9290 GO SUB 9130: LET yy=0: GO SUB z6: LET yy=30: GO SUB z6: LET cal=9280: RETURN
9300 GO SUB 9060: LET tas=9220: LET cal=9400: RETURN
9310 GO SUB 9050: LET cal=9220: LET par=9320: LET lin=9340: RETURN
9320 GO SUB 9020: LET tas=9310: LET lin=9330: RETURN
9330 LET tas=9340: IF k4=0 THEN GO TO 9335
9331 INK 6: PRINT AT 9,20;"RS";AT 10,20;"[redacted]R"
9335 LET mu=1: LET to=0: LET ag=1: LET va=6: LET li=12: GO SUB z1: LET yy=0: GO SUB z6: LET yy=30: GO SUB z6: GO SUB z8: LET xx=11: LET yy=18: GO SUB z4: PRINT AT 16,4; INK 5; PAPER 3;"L[redacted]": LET cal=9320: RETURN
9340 GO SUB 9140: LET lin=9350: LET cal=9310: RETURN
9350 GO SUB 9040: LET par=9360: LET cal=9340: RETURN
9360 GO SUB 9020: LET tas=9350: LET lin=9370: RETURN
9370 GO SUB 9060: LET cal=9360: LET tas=9380: RETURN
9380 LET tas=9390: LET par=9370: LET to=1: LET ag=0: LET mu=0: LET tol=10: LET ra=5: LET zo=24: INK 2: PRINT AT 0,0;"JKFKFG";AT 1,0;"KJFGE";AT 2,0;"GEH": GO SUB 9600: PRINT INK 3;AT 16,0;b$: GO SUB 9950: RETURN
9390 IF k5=0 THEN GO TO 9395
9391 INK 6: PRINT AT 14,3;"RS";AT 15,3;"[redacted]R"
9395 LET par=9380: LET ag=1: LET to=0: LET mu=1: LET va=7: LET li=16: GO SUB z1: LET yy=0: GO SUB z6: GO SUB z7: PRINT AT 16,8; INK 5; PAPER 3;"L[redacted]": LET xx=11: LET yy=20: GO SUB z4: GO SUB 9950: RETURN
9400 LET ag=0: LET to=0: LET mu=1: LET va=6: LET li=11: LET lin=9300: LET cal=9410: GO SUB z2: LET yy=0: GO SUB z6: LET yy=30: GO SUB z6: LET xx=6: LET yy=15: GO SUB z5: LET xx=11: LET yy=4: GO SUB z4: GO SUB z8: GO SUB 9950: RETURN
9410 GO SUB 9250: LET par=9270: LET lin=9400: RETURN
9510 INK 6: PRINT AT x1,y1;"kl";AT x1+1,y1;"mn";AT x1+2,y1;"uv": GO SUB 5000: RETURN
9550 INK 6: PRINT AT x1,y1;"_£";AT x1+1,y1;"ab";AT x1+2,y1;"cd": GO SUB 5000: RETURN
9600 GO SUB 8950: INK 7: PRINT AT 2,19;".";AT 2,24;".";AT 5,25;".";AT 1,12;".";AT 3,7;".";AT 6,10;".";AT 7,7;".";AT 4,15;".";AT 5,21;".": GO SUB 8900: INK 2
9610 PRINT AT 0,26;"CDKFDK";AT 1,27;"CGECD";AT 2,31;"E";AT 3,16; INK 3;"[\";AT 4,16;"]↑"
9620 RETURN
9700 LET di=di+1: GO SUB 8950: PRINT AT 20,18; INK 4;di: GO SUB 8900: RETURN
9950 IF gg=1 THEN INK 6: PRINT AT x1,y1;"kl";AT x1+1,y1;"mn";AT x1+2,y1;"uv": RETURN
9960 INK 6: PRINT AT x1,y1;"_£";AT x1+1,y1;"ab";AT x1+2,y1;"ij": RETURN
9980 GO SUB 8950: LET pu=pu+5: PRINT AT 20,2; INK 6;pu: GO SUB 8900: RETURN
```

LISTADO 2

Linea	Datos	Control
1	FFFC9BF7EB9C77FFE17F	2026
2	9F67DB3CDBFF8FFCFF8F	1808
3	677B03FFF09FFFFD8FFF	1789
4	F3FFFFC3FFFF3BF18FFF	2204
5	0243E5F3B474E5D48086	1540
6	4F9F5ED35FCE7764F574	1424
7	A2F2E2C154DB5FCD9E83	1715
8	8F07DC77190606030301	533
9	DFDDBDD6F77B7AB6EF73	1875
10	00000000000008FBDDBEF	838
11	6EB59DEFBB7FD6987040	1543
12	C080DD7D6A2E36140C08	912
13	00FFAAAAAAFF0000EF6B	1366
14	753D362F6DDFFBBAD65D	1355
15	777ADAF7FF7F3F3FDF0F	1452
16	8338FFFFFEFEFCF8E308	1940
17	000071B7DA7DBBC3C00E0	1017
18	78AE66DB3D3C00307864	1004
19	FCE08000000C1E263F07	754
20	01000010204081020408 0	256
21	80C0E0F0F8FCFEFF0103	1797
22	070F1F3F7FFF03030307	514
23	07070F0FC0C0C0E0E060	1164
24	70B00F1F1F1F3F3F3F7F	712
25	B0B8D8D8DCECECFE0055	1823
26	007F7F7F7F7F005400FE	973
27	F6F6F6F6071F3F7F7FFF	1594
28	FFFFE0F8FCFEFEFFFFFF	2507
29	FFFFFF7F7F3F1F07FFFF	1630
30	FFFEFEFCF8E00030C1214	1540
31	14100C03C8721819141D	463
32	70C0306A752077A34006	959
33	06AF4F06E3C60CE00E0F	956
34	0600061E003E7070200C	372
35	060D1A3406030300010F	125
36	001FE0604010708002850	839
37	06030007000D0003F060	368
38	80A020F000E00E0C0E02	826
39	345E2E141C1C0C020306	291
40	0D0A134E18982080E03	537
41	C03048282808 30C060F5	981
42	F260C76330070C56AE04	967
43	EEC502600E0E24306060	917
44	582C70F06000607800 7C	920
45	070602080E0D140A60C0	368
46	C00080F000F80F060105	835
47	040F000760C000E00080	714
48	00C03838304 0C060B050	960
49	703070402C7A74000000	618

DUMP: 40.000
N.º BYTES: 487



Todos los programas publicados por MICROHOBBY están también disponibles en cinta de cassette para ahorrarte el fatigoso trabajo de copiarlos.

TRUCOS

PROTEGER Y DESPROTEGER

Aun cuando están muy de moda las protecciones en los programas, a cada cual más sofisticadas, es muy corriente encontrarse de nuevo con algunas de las utilizadas en los primeros tiempos.

Nuestro amigo Joan Peñalver nos envía desde Beges (Barcelona) un truco muy sencillo que podemos utilizar en nuestros propios programas.

Si utilizamos POKE 23730,PEEK 23730-5,0, desactivaremos el ANTI-BREAK.

Y si, por el contrario, utilizamos POKE 23613,PEEK 23730-3 activamos el BREAK.

ROM MUSICAL

Muchas veces nos quebramos la cabeza intentando crear una tabla de variables aleatorias, pero, sin duda, no se nos habrá ocurrido utilizar los valores que contiene la propia ROM.

Para muestra, no tenéis más que teclear el listado que os ofrecemos, donde se utiliza el contenido de las celdillas de la ROM comprendidas entre la dirección 0 hasta la 16383, para generar un sonido aleatorio.

IN THE IN

En alguna que otra ocasión hemos ofrecido programas que utilizan para controlar el movimiento la función IN; esto ha ocasionado muchos problemas a los usuarios de Spectrum, sobre todo porque según las distintas versiones esta función devuelve un valor diferente.

Para solucionar este posible problema, es conveniente introducir al principio del programa una línea como la que sigue:

```
LET tecla=64*(IN 256=255)
```

Y en todas las líneas donde se pregunte por un IN, poner éste y luego la cantidad entre paréntesis seguida de tecla. Es decir, que si fuese el IN 65500, el que estamos analizando, deberemos poner:

```
(IN 65500)-tecla
```

ESPIRAL

Desde un pueblo de Badajoz, Don Benito, hemos recibido un truco enviado por Carlos Quintana, con él se crea una espiral de puntos.

En la línea 10 podemos cambiar el valor por cualquier número comprendido entre 10 y 80, consiguiendo aumentar o disminuir el espacio entre cada vuelta de la espiral.

Para aumentar la densidad cambiarenos, en la línea 40, el —1 por —01.

Y si queremos poner o quitar puntos de la espiral, lo haremos en el bucle FOR-NEXT de la línea 20, sustituyendo el 700 por otro valor.

```
10 LET radio=80
20 FOR f=0 TO 700
30 LET a=(RND*7)+1
40 LET radio=radio-.1
50 PLOT 128+radio*SIN (f/32*PI
),88+radio*COS (f/32*PI)
60 NEXT f
```



CURIOSOS RANDOMIZES

Además de felicitarnos las Fiestas, Luis Celay Morales, de Leganés (Madrid), nos envía unos cuantos Randomizes que se unen a la ya inmensa lista que hemos venido publicando.

Teclear el siguiente listado y después hacer **RUN 2;** pulsar muchas veces la tecla ENTER, hasta que aparezcan bastantes filas de interrogaciones (?), borrar el RUN 2 con la tecla **DELETE,** y llegar con el cursor hasta la última interrogación. Intentar borrarla y veréis qué pasa.

Por último, nos manda otros dos Randomizes para completar la actuación:

RANDOMIZE USR 1234
RANDOMIZE USR 1267

```
10 FOR m=5000 TO 6000
20 RANDOMIZE USR m
```

CANCIÓN DE NAVIDAD

Cristina Castellano, de Sabadell, nos envía un programa que genera un alegre villancico de Navidad, que podemos utilizar para hacer unas spectrunavidades más alegres.

El programa está totalmente realizado en Basic, pero automáticamente, al terminar de leer las datas, vuelve a empezar la música.

```
 10 REM  CANCION DE NAVIDAD
 20 FOR C=-12 TO 0 STEP 12
 30 READ N,D
 32 BORDER 7
 40 IF NOT N THEN RESTORE : PAU
SE 50: GO TO 70
 50 BEEP D/3,N+C
 52 BORDER 2
 60 GO TO 30
 70 NEXT C
 80 REM TABLA DE NOTAS
 90 DATA 16,4,17,1,16,1,15,1
100 DATA 16,1,17,4,18,1,19,3
110 DATA 21,2,23,1,24,1,26,1
120 DATA 24,1,23,1,21,1,19,6
130 DATA 12,1,14,1,16,2,16,2
140 DATA 16,1,21,2,19,1,12,2
150 DATA 12,2,12,1,19,2,17,1
160 DATA 16,4,17,1,16,1,14,1
170 DATA 12,1,14,6,16,4,17,1
180 DATA 16,1,15,1,16,1,17,4
190 DATA 18,1,19,3,21,2,23,1
200 DATA 24,1,26,1,24,1,23,1
210 DATA 21,1,19,6,12,1,14,1
220 DATA 16,2,16,2,16,1,21,2
230 DATA 19,1,24,6,12,1,14,1
240 DATA 16,2,16,2,21,1.5,11,.5
250 DATA 11,1,11,1,12,7,0,0
```

En este espacio también tienen cabida los trucos que nuestros lectores quieran proponer.

Para ello, no tienen más que enviarlos por correo a MICROHOBBY, Ctra. de Irún km 12,400 28049 Madrid.

ERBE Software

TE OFRECE LO MEJOR DE LOS MEJORES

Para que lo tengas fácil. Aquí tienes una selección de los mejores juegos que puedes encontrar. Cualquiera de ellos tiene la garantía de calidad y adicción que esperas encontrar en un videojuego. Para no llevarte "sorpresas"... elige un juego distribuido por ERBE en cualquiera de las buenas tiendas de informática.

GREMLIN



TOP-GUN (Idolos del Aire)

Top Gun te coloca en la cabina de un TOMCAT F-14. Gráficos vectoriales y la pantalla dividida permiten a 1 ó 2 jugadores luchar entre ellos o contra el ordenador. Tus armas en este combate, no apto para cardiacos, son misiles guiados por el calor, y una ametralladora de 20 mm.



AVENGER (Way of The Tiger II)

Primero fue "Way of The Tiger" en donde tuviste que demostrar tus habilidades para convertirte en Ninja. Ahora es "Avenger", en donde tendrás que demostrar que, además de fuerte, eres hábil e inteligente para conseguir vencer al Gran Guardián. Buena suerte... sólo los bravos sobrevivirán.



GREAT ESCAPE (Gran Escapada)

Alemania 1942. La guerra ha estallado y tú has sido capturado y condenado en un campo de concentración. Tu deber es escapar, pero no te resultará fácil. Necesitarás planear cuidadosamente las acciones utilizando toda tu astucia para escapar con vida del campamento.



MOVIE

Por fin, la primera edicción de Los Justicieros del Software ha llegado a su culminación. A lo largo de este año han desfilado por nuestra revista todos aquellos programas que, por uno u otro motivo, han destacado del panorama del software mundial. Evidentemente, no están todos los que son, pero sí que son todos los que están.

Durante el transcurso de los últimos doce meses, un total de 33 programas han pasado bajo la mirada atenta de nuestros expertos justicieros, quienes han expresado su imparcial y desinteresada opinión a cerca de estos títulos pertenecientes a otras veinte compañías de software diferentes.

Es posible que no todos estéis de acuerdo con los resultados y puntuaciones obtenidas por cada uno de los programas, pero de lo que no cabe duda es de que las opiniones vertidas por estos Justicieros del Software, corresponden a las de auténticos conocedores del tema (recordamos que cada uno de ellos tuvo que responder a un difícil cuestionario), por lo que su veredicto es tan válido como el que más.

Ahora ya sólo nos queda dar la enhorabuena al ganador y agradecer la colaboración de las compañías de software y distribuidoras, sin las cuales no hubiera sido posible la edición de este concurso.

Del mismo modo, también queremos agradecer a todos nuestros lectores, y en especial a los que fueron elegidos como Justicieros del Software, por su masiva y calurosa participación.

Por otra parte, MICROHOBBY manifiesta su deseo de dar continuidad a este premio, por lo que en semanas próximas se convocará una nueva edición de Los Justicieros del Software, para la cual esperamos vuestra colaboración.

Movie: Ficha Técnica

Título original: *Movie*

Programadores: *Diske y Mario*

Compañía: *Imagine*

Distribuidora: *ERBE*

Versiones: *Spectrum y Amstrad*

Tipo de juego: *Videoaventura*

Sinopsis: *Nueva York, años 30. El superdetective Jack Marlow se ve envuelto en una peligrosa misión en la que debe introducirse en la guarida del gángster Bugs Malloy y localizar una cinta grabada que contiene una importante información. Las mujeres, el alcohol y las pistolas juegan un importante papel en el desarrollo de la historia.*

Características: *Diseño tridimensional. Iconos para opciones de movimiento y acciones. Posibilidad de dialogar con otros personajes.*

Lo mejor: *Excelente ambientación. Minuciosidad en los diseños. Fácil manejo de los iconos. Originalidad y sagacidad en los diálogos.*

Lo peor: *Ajeno al programa: el no haber podido ser traducido al castellano.*



PROGRAMA DEL AÑO

N.º	PROGRAMA	COMPAÑÍA	PUNT.
1	Movie	Imagine	9,3
2	Batman	Ocean	9,1
3	Camelot Warriors	Dinamic	9
4	Three weeks in paradise	Mikro-Gen	8,8
5	The way of the tiger	Gremlin Grafics	8,7
6	Olé, Toro	Dinamic	8,5
	Gunfright	Ultimate	8,5
	Green Beret	Imagine	8,5
9	Antiriad	Palace Software	8,3
	Sir Fred	Made in Spain	8,3
11	Saboteur	Durell	8,2
	Rasputín	Firebird	8,2
13	Las tres luces de Glaurung	Erbe	8,1
14	The great escape	Ocean	8
15	Phantomas I	Dinamic	7,9
	Equinox	Mikro-Gen	7,9
17	Cyberun	Ultimate	7,8
	Cauldron II	Palace Software	7,8
19	Leyend of amazon women	Silver Soft	7,6
	Pyracurse	Hewson	7,6
21	Asterix	Melbourne House	7,5
	West bank	Dinamic	7,5
	Fairlight	The Edge	7,5
	Yabba dabba doo	Quicksilva	7,5
25	Infiltrator	U.S. Gold	7,4
26	Dragon's lair	Software proyects	7,2
27	Jack the nipper	Gremlin Grafics	7,1
	I of the mask	Electric dreams	7,1
29	Phantomas II	Dinamic	6,9
30	Mickie	Imagine	6,8
31	Fighting warrior	Melbourne house	6,7
32	Marsport	Gargoyle games	6,6
33	Popeye	DK'Tronics	6,5

LOS JUSTICIEROS DEL SOFTWARE

1.º TRIMESTRE

Justo Soria Gil (Barcelona)
Javier Redondo Girondo (Madrid)
Gorka Polite Villamayor (Pamplona/Navarra)
José Carlos Núñez Cagigal (Santander/Cantabria)
Daniel González (Madrid)
Fco. Javier Valero Sansano (Elche/Alicante)
Celia Menéndez Cortés (Madrid)
Purificación Hernández Arribas (Burgos)

2.º TRIMESTRE

David de Llodio (Zumaya/Guipúzcoa)
Pilar Arias Valcarce (León)
José Antonio Galiana Ballesteros (Cádiz)
Gabriel Martí (Calella/Barcelona)
Juan Markessinis Aliaga (Valencia)
Ricardo Alonso Villemur (Gijón/Asturias)
Roberto Carlos Alonso Fernández (Valladolid)
Francisco Javier Cano Peño (Madrid)

3.º TRIMESTRE

José Luis Segarra Estébanez (Aguilar de Campoo/Palencia)
Pedro Revilla Ortega (Burgos)
Luis García Sánchez (Pto. Sta. María/Cádiz)
Santiago Colas Herrero (Zaragoza)
Ángel Pérez Barreiro (Sabadell/Barcelona)
Mario López Melinat (Marchamalo/Guadalajara)
Misericordia Pagues Gil (Reus/Tarragona)
Alberto A. Cifrián Miranda (Torrelavega/Cantabria)

4.º TRIMESTRE

Alberto Torres Albiñana (Gerona)
Javier Cabrera Peset (Valencia)
Carmen Manzanera Valero (Manises/Valencia)
Luis A. Álvarez (Alcorcón/Madrid)
Juan Carlos Espinel Hurtado (Sevilla)
Xavier Masip Pesquer (Barcelona)
José Antonio Rodríguez Ovalle (Villafranca del Bierzo/León)
Nuria Valdés Díaz de Tuesta (Vitoria/Álava)

APRENDE DE TUS ERRORES

Esta semana veremos tres errores que, si bien no se presentan con frecuencia, sí resulta útil conocerlos ya que nos indican algunas características del funcionamiento del ordenador que, habitualmente pasan desapercibidas.

Jesús ALONSO RODRÍGUEZ

7 RETURN without GOSUB

SIGNIFICADO: «RETURN sin GOSUB». Indica que el intérprete se ha encontrado un comando «RETURN», pero no sabe a dónde retornar, ya que no ha habido un «GOSUB» previo que lo enviara aquí. El hecho de que el error sólo pueda ser producido por el comando «RETURN» puede hacer pensar que su solución es fácil. Nada más alejado de la realidad. En algunos casos puede ser uno de los errores más difíciles de resolver.

CAUSA: Cada vez que el intérprete se encuentra con una sentencia «GOSUB», mete la dirección de la sentencia siguiente en una pila denominada «Pila de GOSUB» y salta a la línea que se le indique. Posteriormente, al encontrar una sentencia «RETURN», lee de la pila la dirección a donde debe retornar, que será la sentencia siguiente al «GOSUB» que le trajo a esta rutina. Como la pila de GOSUB es del tipo «Último en entrar-Primero en salir», permite la anidación de subrutinas, de forma que siemore se retornará a la sentencia siguiente a la última llamada. Pero, ¿que ocurre si se encuentra un «RETURN» si que haya habido una llamada previa con «GOSUB»? En ese caso, cuando el intérprete vaya a buscar la dirección de retorno en la pila, la encontrará vacía y no sabrá sónde retornar, por lo que detendrá la ejecución del programa e imprimirá el mensaje: «RETURN without GOSUB». La causa más frecuente es que se haya saltado con «GOTO» en medio de una subrutina. Hay que tener en cuenta que si hay varias subrutinas anidadas, el error puede no ser detectado inmediatamente, puesto que la pila de GOSUB no estará vacía; sin embargo, el error hará que la pila tenga un elemento de menos y los retornos no se harán en el orden correcto. En cualquier caso, el informe «RETURN without GOSUB» se producirá cuando el programa intente realizar el último retorno, a menos que la alteración producida en la secuencia de retornos, provoque algún otro error con anterioridad.

SOLUCIÓN: En general, se trata de un error sumamente difícil de localizar sin ayuda de un trazador (el comando «TRON» comentado en el primer capítulo y del que carece el Spectrum), por lo que lo más indicado es hacer un organigrama (diagrama de flujo) del programa y ver si la codificación que hemos realizado sigue fielmente los pasos indicados por el organigrama. Este tipo de errores suelen ocurrir en programas con una secuencia de ejecución muy «enmarañada», por lo que el mejor consejo que podemos dar al programador es que escriba sus programas con una estructura lo más clara posible y evite el uso de «miles de GOTOs» (uno de los principales «vicios» que se adquieren aprendiendo a programar en Basic) que hagan difícil seguirle la pista. Si se sigue este consejo, se facilitará mucho la depuiración de cualquier error.

8 End of file

SIGNIFICADO: «fin de fichero». Indica que se ha intentado leer más allá del final de un fichero de acceso secuencial en Microdrive, RS-232 o ZX-Net. Es decir, se han intentado leer más datos que los que contenía el fichero.

CAUSA: este error sólo puede presentarse si se tiene conectado el Interface 1, ya que es el único dispositivo de Sinclair para Spectrum que permite manejar ficheros de acceso aleatorio. Cuando se abre uno de estos ficheros y se asigna a una corriente (por ejemplo: con OPEN #5 ...) todo lo que se escriba por esa corriente (PRINT #5 ...) irá a parar al fichero. Asimismo, lo que se lea de esa corriente (INPUT #5) será leído del fichero. Si cuando se intenta abrir el fichero, éste no existe, se abrirá para escritura, en caso contrario, se abrirá para lectura. Pues bien, si con un fichero abierto para lectura, ejecutamos más INPUTs que los PRINTs que se ejecutaron al crearlo, no habrá datos que leer y el programa se detendrá con este error.

SOLUCIÓN: lo más probable es que se haya utilizado un bucle para escribir los datos en el fichero y otro para leerlos. Convendría comprobar si ambos bucles tienen el mismo número de interacciones. Por otro lado, puede ocurrir que se intente leer, por segunda vez, un fichero sin haberlo cerrado y vuelto a abrir, éste sería otro posible motivo que provocaría el error «End of file».

9 STOP Statement

SIGNIFICADO: «sentencia STOP». Indica que el intérprete ha detenido la ejecución porque se ha encontrado con una sentencia «STOP» que así se lo ordena.

CAUSA: no se trata, propiamente, de un error. Si el intérprete ha encontrado una sentencia «STOP» es porque el programador la ha puesto ahí por alguna razón. Normalmente, se pone una sentencia «STOP» durante el proceso de depuración de un programa, para detenerlo en ese punto y comprobar el estado de algunas variables. Se puede reanudar la ejecución del programa con el comando «CONTINUE» que empieza ejecutando la sentencia siguiente al «STOP».

SOLUCIÓN: cuando el programa ha puesto ahí la sentencia «STOP», será por alguna razón. La solución evidente para que el programa no se detenga es editar el listado y quitar la sentencia «STOP». Puede ocurrir, no obstante, que el programa se detenga sin que estuviera previsto, es decir, que alcance la sentencia «STOP» cuando no debiera haberla alcanzado. En ese caso, nos encontraríamos ante un «error lógico» y, como de costumbre en este tipo de errores, lo mejor es trazar un organigrama y comprobar si el programa que hemos escrito lo sigue fielmente. Siempre es más sencillo detectar los errores lógicos sobre un organigrama que sobre un listado.


YA ESTA A
LA VENTA
EL N.º 5 DE...
Juegos & ESTRATEGIA
VIVE CON SPECTRUM LA
BATALLA MAS APASIONANTE
DE ESTA DECADA.
¡APROVECHA NUESTRA OFERTA!
PONCE
3 JUEGOS & ESTRATEGIA POR EL PRECIO DE 2
(sólo 2.250 ptas.)
DISPONIBLE PARA SPECTRUM 48 K, 128 K, +2 K
MALVINAS 82
Recorta o copia este cupón y envíalo a Hobby Press, Apartado de Correos 232. Alcobendas (Madrid)
☐ Sí, deseo recibir en mi domicilio el número 5 de Juegos & Estrategias, "Malvinas 82", al precio de 1.125 ptas.
☐ Sí, deseo recibir en mi domicilio tres ejemplares de Juegos & Estrategia, y pagar sólo dos (2.250 ptas.).
Esta oferta es válida sólo hasta el 20 de febrero de 1987.
Los juegos que deseo son
La versión que elijo es para: ☐ Spectrum ☐ Amstrad ☐ Commodore
Nombre
Apellidos
Domicilio
Localidad
C. Postal
Fecha de nacimiento
Teléfono
Para agilizar tu envío es importante que indiques el código postal.
Forma de pago: ☐ Mediante talón bancario a nombre de Hobby Press, S.A.
☐ Mediante giro postal a nombre de Hobby Press, S.A. n.º
☐ Mediante tarjeta de crédito número ☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐
☐ Visa ☐ Master Charge ☐ American Express
Fecha y firma
No se admiten solicitudes de cintas contra reembolso.

LO NUEVO

DESAFÍO EN EL CASTILLO DE LAS S

Uno de los videojuegos más populares de los últimos años viene a hacer su fulgurante aparición en el Spectrum. Su fama ya ha alcanzado muy importantes cotas, pero seguro que a partir de este momento, y gracias a su excelente conversión al ordenador, se va a convertir en uno de los grandes éxitos del año.





GAUNTLET

Arcade

U.S. Gold

Si hay algún programa que cuando sale comentado en nuestra revista, la gran mayoría de los usuarios ya tienen conocimientos de sus características y argumento principal, Gauntlet es, sin duda, uno de ellos.

A pesar de que este juego no ha tomado como protagonista a ningún personaje famoso del mundo del cine o del cómic, o de que su tema no está basado en ninguna película de actualidad, seguramente una buena parte de vosotros habéis reconocido su nombre nada más verlo.

Este juego, en su correspondiente y genuina versión para las máquinas de videojuegos, es ya conocida en todo el mundo, lo que de por sí ya es una buena garantía de calidad.

Al menos Gauntlet trae excelentes antecedentes; pero ¿cómo ha quedado el juego para Spectrum? Esta es una pregunta que en las próximas líneas trataremos de desvelar, aunque, por si sois impacientes, os podemos adelantar que, a nuestra manera de ver, ha quedado un juego sensacional. De todas formas, si lo que os apetece es conocer más a fondo las características del programa, no paséis aún la página que quedan muchas cosas que contar.

De momento, empezaremos comentando cuál es el argumento en el que se basa este Gauntlet.

Nos encontramos en el Castillo de las Sombras, lugar infecto y peligroso, donde moran todo tipo de fantasmas, demonios y demás individuos de malas intenciones y dudosa reputación.

Hemos dicho «nos» no con afán de florituras literarias, sino porque realmente el juego está protagonizado por varios personajes. A saber: Thor, Thyra, Merlín y Questor.

Thor, el guerrero, es el cachas del equipo, y, tanto en la lucha cuerpo a cuerpo como en lo que al poder de disparo se refiere, sus condiciones son excelentes.

Thyra, la parte sensual del juego; a pesar de su condición femenina también posee unas buenas dotes de lucha, aunque su mayor virtud la obtiene de su habilidad en el dominio de la espada.





Merlín, el mago. Su carencia de fuerza la suple con sus increíbles dotes mágicas. A pesar de su edad es un buen guerrero.

Questor, el enano, representa la astucia y aunque no posee una fuerza excesiva, es muy práctico en el combate.

Como veis, cada uno de los protagonistas posee unas determinadas condiciones físicas y mentales. El desarrollo del juego dependerá, pues, del personaje que controlemos.

SOMBRAS







Al principio del juego, se nos pide que elijamos entre uno de estos cuatro luchadores, por lo que según deseemos que se desarrollen los acontecimientos, tendremos que decidir con cuál de ellos quedarnos.

No penséis, sin embargo, que nos encontramos ante un juego de estrategia. Nada más lejos de la realidad. Gauntlet es un programa con el desarrollo típico de un arcade de acción; es decir, que dejando a un lado estas pequeñas variaciones que se pueden producir, nuestra misión principal va a consistir en eliminar a todo bicho viviente que se interponga en nuestro camino.

Estas variaciones a las que hemos hecho alusión son las que se refieren a la capacidad de eliminar con mayor o menor facilidad a determinado tipo de enemigos. En la práctica, elijamos al personaje que elijamos, lo que vamos a tener que hacer es disparar continuamente durante todo el desarrollo del programa y tratar de acabar con el mayor número posible de fantasmas.

Sin embargo, aparte de este objetivo fundamental, otros factores importantes también tienen lugar en el desarrollo del juego. Por ejemplo, existen repartidos entre los muros y pasillos de este tétrico castillo un buen número de amuletos, llaves, alimentos, trampas o transportadores. Como podéis suponer, cada uno de estos ítems tiene su propia utilidad y sin ellos es imposible avanzar en la misión. Pero también hay que señalar que sus aplicaciones son, en la mayoría de los casos, inmediatas, por lo que no tendremos que preocuparnos de cuándo y en qué lugar utilizarlos: nada más cogerlos sabremos sus efectos.

Otro detalle muy importante de Gauntlet es que pueden participar simultáneamente dos jugadores. De esta forma se consigue dividir el riesgo y multiplicar la diversión. Los dos personajes pueden ayudarse el uno al otro, por lo que el número de enemigos a eliminar se reparte entre ambos y resulta mucho más rápido y sencillo el llegar más lejos en la misión, a la vez que todo el desarrollo de la acción resulta mucho más entretenido y adictivo.

Sin duda alguna, Gauntlet es un programa que divierte a todos. Esto lo logra principalmente por el dinamismo y rapidez de su acción. A pesar de que todo el juego se desarrolle a un ritmo muy vivo, no es el típico arcade en el que te eliminan al menor descuido, sino que, por el contrario, puedes moverte con relativa facilidad por las pantallas y seguramente podrás conocer muchas de ellas antes de ser «borrado del mapa». De esta forma, se combina a la perfección la emoción de tener que movernos rápidamente, con la tranquilidad de que no vamos a morir a la primera de cambio.

Quizá el aspecto más controvertido que presenta Gauntlet es la escasa brillantez de sus diseños. La verdad es que éstos son de una gran simplicidad y la mayoría de las pantallas están formadas por líneas más o menos decoradas que representan los muros y pasillos del castillo, aunque, eso sí, todas ellas están literalmente «plagadas» de pequeños personajes a eliminar.

Pero, por otra parte, esta sobriedad en los gráficos es más que comprensible si consideramos que existen aproximadamente unos 100 niveles diferentes en los que se desarrolla la acción, estado compuesto cada uno de ellos por otras 10 o más pantallas. El número de éstas, como podréis deducir, asciende a una cifra más que considerable.

En lo que se refiere al scroll de las pantallas, también hay que señalar que está realizado con una gran suavidad, ya que apenas te das cuenta de que éste se está produciendo.

En definitiva, podemos decir que quizá en un principio el programa defrauda ligeramente por su sobriedad gráfica, pero que esto se ve sobradamente suplido por la enorme cantidad de diversión que nos ofrece su desarrollo. Gauntlet es, sin duda, uno de los juegos más adictivos de cuantos se pueden encontrar en el mercado. Un auténtico «juegazo» que entusiasmará a todos.

Por último, diremos que debido a su enorme cantidad de pantallas y escenarios diferentes, todo el programa no puede ser cargado en un solo bloque, por lo que, a medida que consigamos superar los diferentes grupos de niveles, tendremos que poner de nuevo en marcha la cinta e ir grabándolo por partes. Y es que no hay nada perfecto.



EL UNIVERSO EN PELIGRO





UNIVERSAL HERO

Videoaventura

Mastertronic

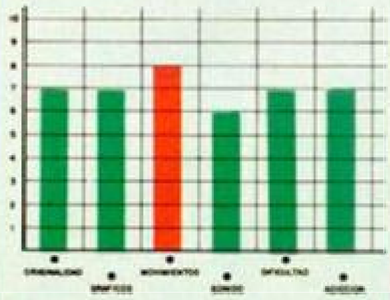
Siguen apareciendo programas de los llamados serie barata, que la casa Mastertronic está comercializando actualmente. Algunos de estos programas son antiguos títulos que fueron lanzados al mercado hace algún tiempo y que se han vuelto a poner en circulación, cosa que los nuevos usuarios del Spectrum, llegarán a agradecer algún día.

El programa está ambientado en un planeta imaginario, similar a la luna, en el que tenemos que buscar una serie de objetos, nueve en total, para poder salvar al planeta y con ello nuestra propia vida. Estos objetos pueden ser llevados todos a la vez y posteriormente ser utilizados a nuestra conveniencia. Hasta llegar a entrar en la sala de la computadora, y arreglar nuestra nave de carga para poder escapar con los objetos del planeta, deberemos luchar incansablemente por estos tortuosos y peligrosos parajes.

Los gráficos del juego son de gran calidad, obteniendo un buen efecto de movimiento del protagonista, así como los pululantes bichitos que nos impiden atravesar cómodamente las pantallas del intrincado laberinto que es este planeta.

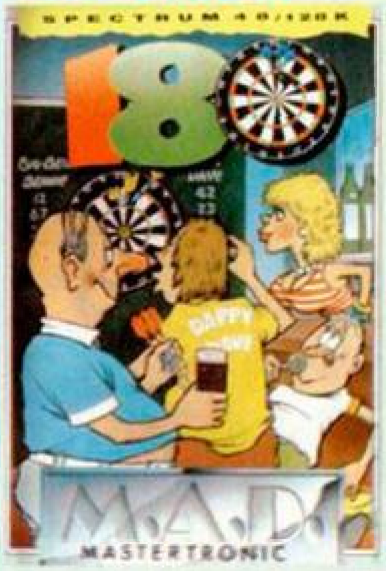
El manejo del juego quizás resulte algo complicado por la incómoda colocación de las teclas, pero, tras algunas partidas, acabaremos adaptándonos a su uso.

El juego en sí nos recuerda a otros muchos programas de similares características y adicción. Un programa entretenido, pero poco más.





DA EN LA DIANA



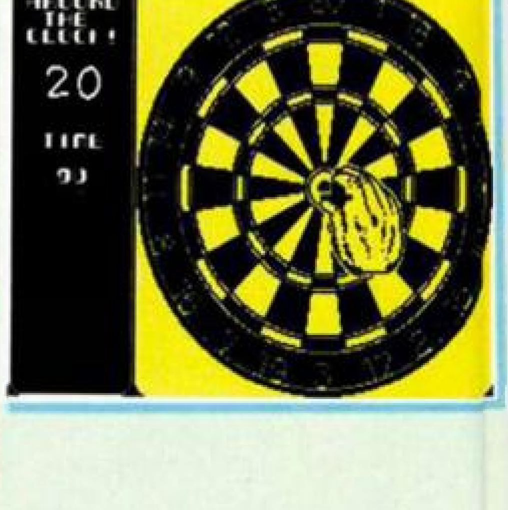
180

Estrategia

Mastertronic

Este 180 es el típico juego de dardos que tanto gusta a los ingleses y que los españoles solemos practicar en el pub del barrio, aunque este programa nos evita el riesgo de saltarnos o saltar un ojo a alguno de nuestros contrincantes.

Con este juego se puede hacer una competición, ya que el programa posee una tabla de récords en que se inscribirán los mejores.





El juego no es tan sólo una diana y unos dardos, ya que además podemos jugar con los distintos personajes del programa. La primera escena que vemos es una diana y una mano flotando a la cual nosotros dirigimos. Pero al lanzar nuestros tres dardos, cambia la pantalla y aparece una barra de un bar en la que una gorda señora está continuamente llenando vasos de cerveza y deslizándolos por la barra. Para darle más realismo en este pub no está prohibida la entrada de perros y de vez en cuando un can aparece olisqueando por la barra y al llegar a uno de sus extremos este lindo animalito hace algunas de sus necesidades biológicas.

En el juego entre dos jugadores no aparece la barra del bar por lo que es bastante más rápida (algo muy de agradecer, pues cuando se juntan muchos jugadores en una competición la cosa se puede eternizar).

Los gráficos y la calidad del juego en sí son extraordinarios, y su gran realismo nos hará pasar buenos ratos delante del ordenador, lanzando tan particulares dardos.



UNA FUGA DE LOCURA



DUSTIN

Videoaventura

Dinamic

Entre las últimas creaciones de la casa Dinamic nos encontramos con el programa Dustin.

El juego está basado en la fuga de una cárcel de alta seguridad, en la que se encuentra prisionero nuestro protagonista, Kid Saguf, conocido por el sobrenombre del rey del Robo de Joyas.

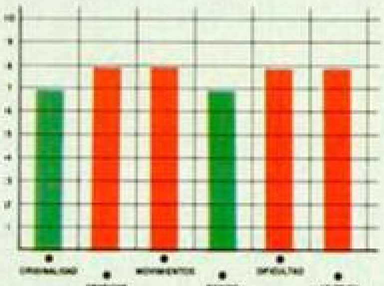
Esta prisión ha sido diseñada para mantener alejados a los esbirros mafiosos, y la mala calaña que anda suelta por la ciudad, de sus tranquilos y apacibles ciudadanos.

Tras largos años de penosa custodia policial, Kid ha conseguido encontrar una posible escapatoria de los muros de la cárcel. Los horarios de los policías ya no son un problema para él, todos tienen un movimiento monótono y rutinario que ha conseguido aprender de tal forma, que podría seguir los pasos de sus guardianes sin necesidad de saber la hora que es ni si es de día ni de noche.

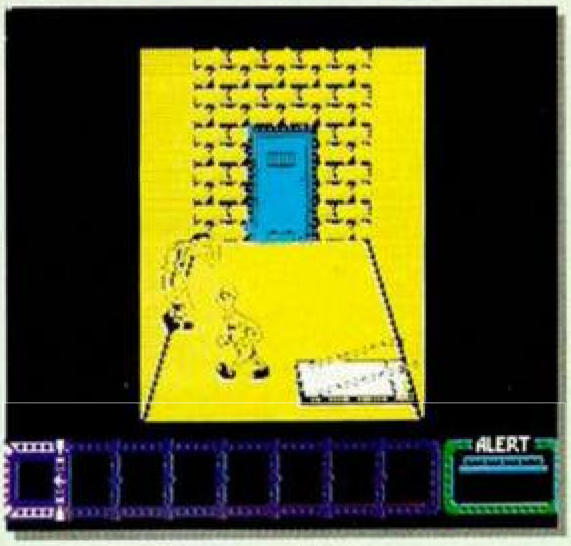
Pero aún le quedan muchas cosas que saber del entorno en el que se encuentra. Hasta las rejas de su celda llega el fresco aroma del mar, pero en ocasiones, cuando el viento sopla del oeste, llegan hasta él ciertos olores campestres, y sonidos de animales como en una cercana selva, que desde sus barrotes no se llega a ver. Dustin anhela la huida, para lo cual deberá conseguir ciertos objetos que poseen los policías y los demás presos, valiéndose de toda la astucia y fuerza física que sea capaz de desarrollar.

Aún hay muchas cosas que Kid no ha conseguido por sí solo, y para no perder la costumbre tendremos que ayudarle a cruzar los muros de la penitenciaría, y salvar todos los obstáculos que se le presenten. Esta dificultad nos hará pensar que, al fin y al cabo en la cárcel no se está tan mal, con sus tranquilos y silenciosos pasillos, su agradable comida, y esa siesta a la sombra tan reconfortante.

En suma, el juego es adictivo, por lo que nos serán reconfortantes las horas y horas que tendremos que pasar frente al ordenador, contemplando unos buenos gráficos que en ocasiones nos harán, incluso, alegrar la faz.







FUNCIONES Y PROCEDIMIENTOS EN PASCAL

Hasta ahora solamente hemos visto sentencias del Pascal, pero esto no nos permite apreciar toda la potencia del lenguaje proporcionada por la estructuración y modularidad.
Este artículo es una introducción al tema mediante la declaración de Funciones y Procedimientos.

F. Javier MARTÍNEZ GALILEA

Pascal suministra un gran número de funciones predefinidas, alguna de las cuales vimos en el artículo anterior. Evidentemente, al definir el lenguaje, su autor no pudo preveer todas nuestras necesidades, pero sí nos dejó la puerta abierta para que las definiéramos nosotros mismos.

En realidad, al escribir **SIN (2*PI)** estamos llamando a una función —SIN— y dándole unos valores adecuados —(2*PI)—, que se llaman parámetros reales, para que nos calcule lo que queremos —el seno de (2*PI)—. Cuando definamos a nuestro gusto una función, haremos lo mismo.

Declaración de funciones

Para declarar una *función* necesitamos especificar el nombre por el que deseamos llamarla, el tipo que tendrán sus parámetros y el tipo de resultado esperado. Con un ejemplo se verá mejor:

```
FUNCTION MAYOR (X,Y:INTEGER):INT
GER;
BEGIN
  IF X>Y THEN MAYOR:=X
         ELSE MAYOR:=Y
END;
```

FUNCTION significa que vamos a definir una función.

MAYOR es el nombre que le hemos dado y por el que la llamaremos (la función halla el mayor de dos números enteros).

(X,Y:INTEGER) señala que la función recibirá como parámetros dos enteros.

:INTEGER significa que el resultado de la función, será un entero (evidentemente, el mayor de dos enteros es también un entero).

A la función la llamaríamos, por ejemplo, con **MAYOR(7,9)** y devolvería **9**.

Observaréis que hemos puesto dos variables **(X,Y:INTEGER)** y no las dos constantes de nuestro ejemplo **(7,9)** ya que generalmente reclamaremos a una función desde diferentes lugares del programa y con datos distintos. En cada caso, **X** e **Y** tomarán los valores que se hallen entre paréntesis y por el orden en que se encuentren. O sea, en este caso, **X** toma el valor **7** e **Y** el valor **9**.

X e **Y** se llaman parámetros formales y sólo se usan dentro de la declaración de la función. Como véis no es necesario declararlas en la parte de declaraciones del programa: al hacer **(X,Y:INTEGER)** quedan definidas.

Por supuesto, una función puede contener diferentes tipos de parámetros formales y el tipo del resultado también puede ser diferente. Por ejemplo:

```
FUNCTION ASCII (A:CHAR;X:INTEGER
):BOOLEAN;
BEGIN
  IF CHR(X)=A THEN ASCII:=TRUE
              ELSE ASCII:=FALSE
END;
```

Esta función devuelve **TRUE** si hemos introducido la letra y su código **ASCII** correcto y **FALSE** en caso contrario.

```
ASCII(V,50) ---> FALSE
ASCII(A,65) ---> TRUE
```

(La función predefinida **CHR(X)** donde **X** es entero, devuelve el carácter correspondiente al código ASCII de **X**).

Declaración de procedimientos

Los *procedimientos* tienen una forma parecida a las *funciones*, pero son mucho más flexibles. Mientras que las *funciones* necesitan unos parámetros y siempre proporcionan un resultado, los *procedimientos* pueden tener parámetros o no y pueden dar uno, varios resultados o ninguno. Son como pequeños subprogramas.

La forma de llamar a un procedimiento es similar a la empleada con una función sólo que ahora puede no ser preciso pasarle parámetros. Por ejemplo:

```
PROCEDURE ASTERISCOS;
VAR
  I:INTEGER;
BEGIN
  FOR I:=1 TO 10 DO
    WRITELN('*****')
END;
```

nos escribiría 10 filas de ***** sólo con escribir: ASTERISCOS.

Existen dos tipos de parámetros que se pueden pasar a un procedimiento. Todos los que hemos tratado hasta ahora eran parámetros por valor y su funcionamiento consiste en que al entrar en el procedimiento se asigna a cada parámetro el valor del parámetro real con el que hemos llamado al procedimiento:

```
PROCEDURE EJEMPLO (A,B:REAL);
BEGIN
...
END;

EJEMPLO (3.27*J,2.12)
```

se asigna a **A** el valor de **3.27*J** (**J** se ha asignado en otro lugar y suponemos vale **22.89**) y a **B** el valor **2.12** y se ejecuta el procedimiento con esos valores. Notar que no hemos afectado para nada a los parámetros reales y que **J** sigue valiendo **22.89**.

En cambio, puede ser necesario que un procedimiento reciba un parámetro, obedezca sus sentencias y modifique el parámetro correspondiente al devolverlo. Esto se hace con parámetros variables o por referencia que se definen anteponiendo **VAR** al parámetro en cuestión. Veamos esto con un ejemplo:

```
PROCEDURE EDADES (X,Y,Z,T:INTEG
ER;VAR MAYOR:INTEGER);
BEGIN
  IF X>MAYOR THEN MAYOR:=X;
  IF Y>MAYOR THEN MAYOR:=Y;
  IF Z>MAYOR THEN MAYOR:=Z;
  IF T>MAYOR THEN MAYOR:=T
END;
```

El procedimiento calcula la mayor de las cinco cifras que le pasamos como parámetros. Paso a paso lo que ocurre es lo siguiente:

Llamamos al procedimiento con **MAYOR(PEPE,JUAN,PEDRO,FELIPE,MASVIEJO)** supo-

niendo que esas variables están definidas como:

PEPE,JUAN,PEDRO,FELIPE,MASVIEJO:INTEGER;

y que en algún lugar del programa hemos asignado a estas variables las edades de **PEPE**, **JUAN**, **PEDRO**, **FELIPE** y el **MASVIEJO** de otro grupo:

```
PEPE:=15;
JUAN:=23;
PEDRO:=41;
FELIPE:=12;
MASVIEJO:=30;
```

y deseamos saber quién es el mayor de todos almacenando el valor de su edad en la variable **MASVIEJO.**

La llamada al procedimiento sería equivalente a haber hecho:

MAYOR(15,23,41,12,30); al entrar en el programa se asigna a cada parámetro por valor o por referencia su parámetro real correspondiente, esto es:

```
X:=15; Y:=23; Z:=41; T:=12; MAYOR:=MASVIEJO;
```

Notar que **MAYOR** no ha recibido el valor que tuviera **MASVIEJO,** sino la variable **MASVIEJO** propiamente dicha. Al terminar de ejecutar el procedimiento **MAYOR** tendrá el valor de **Z**, pero como es un parámetro variable, en realidad no asignamos a **MAYOR** el valor de **Z** sino a la variable que representa, esto es, a **MASVIEJO.** Al salir del procedimiento habremos perdido todas las variables del mismo y tendremos lo siguiente:

```
PEPE=15; JUAN=23; PEDRO=41;
FELIPE=12; MASVIEJO=41;
```

Una de las ventajas más importantes que incorpora **Pascal** se refiere a su *modularidad.* Podemos definir variables, tipos e incluso otras funciones o procedimientos dentro de una función o procedimiento dado. Estos elementos definidos dentro de la función o procedimiento sólo se podrán usar dentro de él y sólo ocuparán memoria mientras se ejecute esa parte del programa. Es muy conveniente trabajar así, sobre todo en programas extensos ya que dotamos de total independencia a cada procedimiento y se pueden probar y ejecutar por separado.

Debemos reservar la declaración de variables globales (al principio del programa) sólo para aquéllas que se vayan a utilizar a lo largo de todo el mismo. Así, cada *procedimiento* o *función* se puede «trasladar» de programa sin modificar nada, ni siquiera el nombre de las variables, puesto que son internas a él. Incluso puede haber variables globales y pertenecientes a un procedimiento o función con el mismo nombre, ya que ambas son diferentes.

En el ejemplo hemos introducido una sentencia nueva: **CASE** que simplifica enormemente el uso de **IF...THEN...ELSE** cuando existen numerosas alternativas:

```
CASE A OF        3:WRITE('3');
1:WRITE('1');    4:WRITE('4');
2:WRITE('2');    END;
```

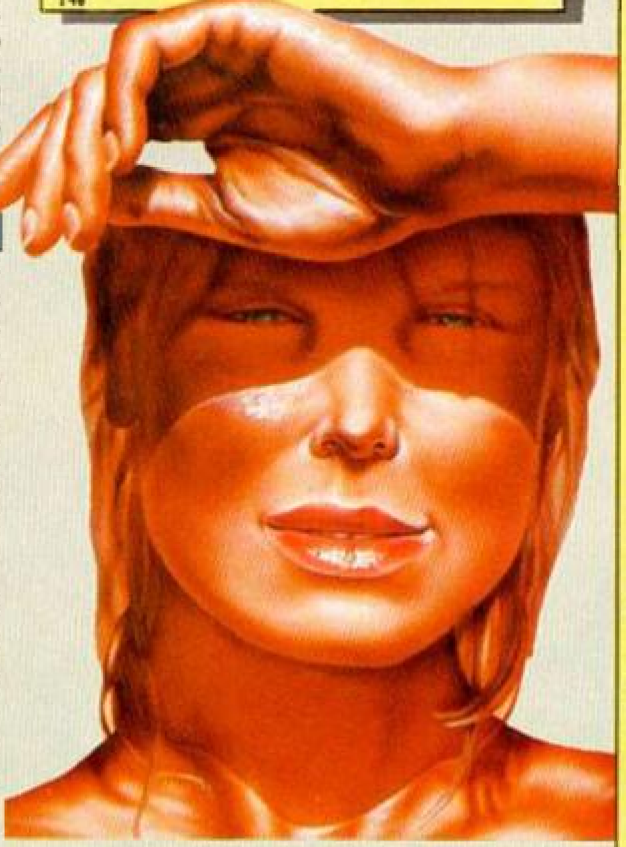
La sentencia funciona como un selector. El programa ve qué valor toma **A** y ejecuta la o las instrucciones que contenga esa etiqueta, saltando a continuación hasta el **END** del **CASE** y continuando el programa. Veremos toda la potencia de esta sentencia más adelante.

El programa propuesto os presenta el calendario completo del año que deseéis, sin más que introducir el día de la semana del 1 de enero. (Podéis sacarlo por impresora, si disponéis de ella, con la opción del **HP4T: CHR(16)**).

Para hacer que el programa calculase por sí mismo el día de la semana del 1 de enero, no haría falta modificar nada, sólo introducir el procedimiento con el algoritmo correspondiente y una llamada desde el programa principal. Ésta es una buena prueba de la modularidad del Pascal. Os retamos a que lo intentéis.

LISTADO CALENDARIO

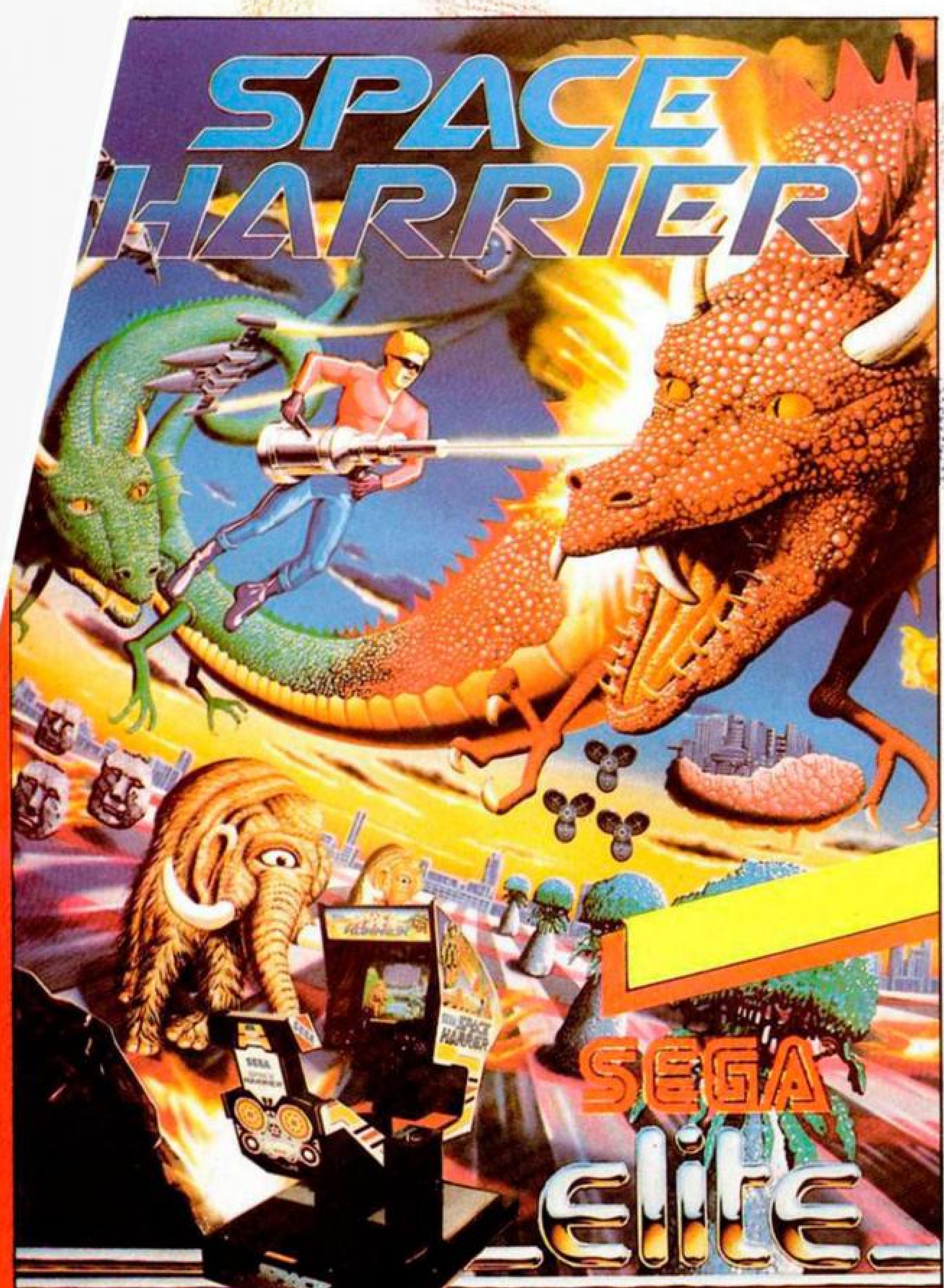
```
10 PROGRAM CALENDARIO;
20
30 TYPE
40   MESES3=ARRAY[1..10]OF CHAR;
50
60 VAR
70   DIASENMES,ANNO,J,ESPACIOS,DIA,DIAPRIMERO:IN
  TEGER;
80   X:CHAR;
90   MES:MESES3;
100
110 PROCEDURE PRESENTACION;
120 VAR
130   E,I:INTEGER;
140
```



```
150 BEGIN
160   PAGE;
170   FOR I:=1 TO 10 DO
180       WRITELN;
190   WRITELN('PROGRAMA GENERADOR DE CALENDARIO');
200   WRITELN;
210   WRITELN;
220   WRITELN('      F.JAVIER MARTINEZ G.');
230   WRITELN('            MICROHOBBY');
240   FOR I:=1 TO 20000 DO E:=E;
250 END; (* DE PRESENTACION *)
260
270 PROCEDURE QUEMES(M:INTEGER;VAR MES:MESES3;VAR
   DIASENMES:INTEGER);
280 BEGIN
290   CASE M OF
300   1:MES:='ENERO     ';
310   2:MES:='FEBRERO   ';
320   3:MES:='MARZO     ';
330   4:MES:='ABRIL     ';
340   5:MES:='MAYO      ';
350   6:MES:='JUNIO     ';
360   7:MES:='JULIO     ';
370   8:MES:='AGOSTO    ';
380   9:MES:='SEPTIEMBRE';
390   10:MES:='OCTUBRE   ';
400   11:MES:='NOVIEMBRE ';
410   12:MES:='DICIEMBRE '
420   END;
430   CASE M OF
440   1,3,5,7,8,10,12:DIASENMES:=31;
450   4,6,9,11:DIASENMES:=30;
460   2:DIASENMES:=28
470   END;
480 END; (* DEL QUEMES *)
490
500 PROCEDURE CALENMES(DIASMES:INTEGER;VAR DIASEM:I
   NTEGER;MES:MESES3);
510 BEGIN
520   WRITELN;
530   WRITELN(MES,'--',ANNO);
540   WRITELN;
550   WRITELN('LUN MAR MIE JUE VIE SAB DOM');
560   FOR ESPACIOS:=1 TO DIASEM-1 DO
570   WRITE ('    ');
580   FOR DIA:=1 TO DIASMES DO
590   BEGIN
600       WRITE(DIA:2,'  ');
610       IF DIASEM=7 THEN
620           BEGIN
630               WRITELN;
640               DIASEM:=1;
650           END
660       ELSE DIASEM :=DIASEM+1;
670   END;
680 WRITELN;
690 END;
700
710 (* PROGRAMA PRINCIPAL *)
720 BEGIN
730   PRESENTACION;
740   WRITE (CHR(12));
750   WRITE('Que anno quieres ? ');
760   READ(ANNO);
770   REPEAT
780       WRITELN('DIA DE LA SEMANA DEL 1 ENERO ?');
790        WRITELN('1-LUNES');
800        WRITELN('2-MARTES');
810        WRITELN('3-MIERCOLES');
820        WRITELN('4-JUEVES');
830        WRITELN('5-VIERNES');
840        WRITELN('6-SABADO');
850        WRITELN('7-DOMINGO');
860       READ (DIAPRIMERO);
870   UNTIL (DIAPRIMERO>=1) AND (DIAPRIMERO<=7);
880   FOR J:=1 TO 12 DO
890   BEGIN
900       QUEMES(J,MES,DIASENMES);
910       CALENMES(DIASENMES,DIAPRIMERO,MES);
920       IF J<12 THEN
930       BEGIN
940           WRITE('Pulse ENTER para otro mes...');
950           READ(X);
960       END;
970   END;
980 END.
```

ZAFI CHIP

¡¡ impresionante !!

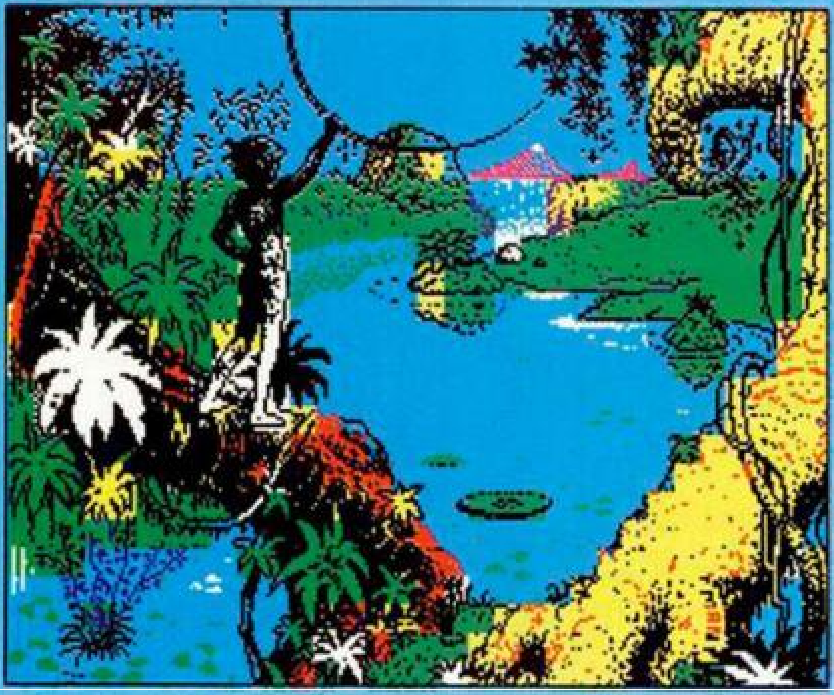


—Spectrum Commodore —
—Amstrad—
—Amstrad Disk —

Ten nervios de acero, alerta tus sentidos y preparate para un viaje terrorifico al futuro.

ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141. 28046 Madrid
Tel. 459 30 04. Tel. Barna. 209 33 65. Télex: 22690 ZAFIR E

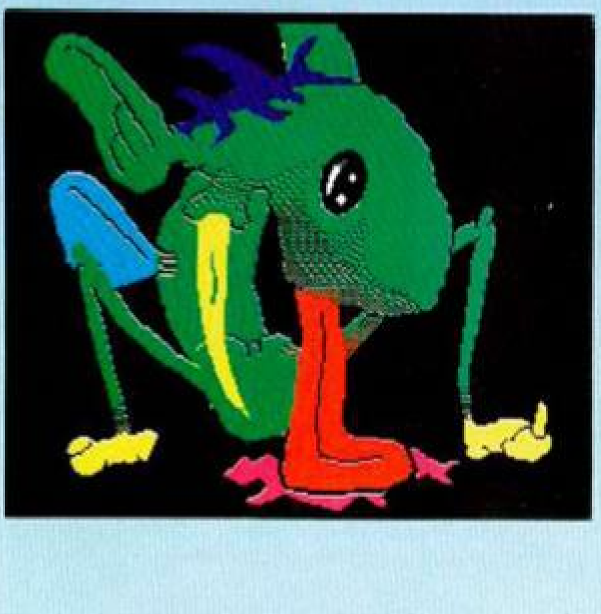
PIXEL A PIXEL



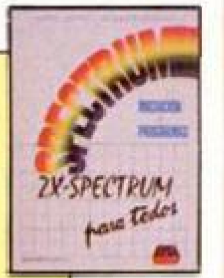
M. A. Naranjo Naranjo
Alcobendas (Madrid). N.° 16. 35 puntos.



J. V. Escares Guijarro
(Valencia). N.° 17. 35 puntos.



J. R. Suárez Pérez
Avilés (Asturias). N.° 87. 26 puntos.



SPECTRUM PARA TODOS

EDICIONES ELISA.
Marcel Henrot.
Jacques Boisgontier
137 páginas.

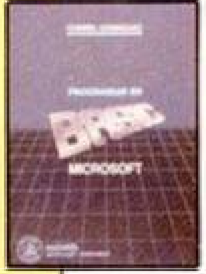
Para dialogar con el ordenador disponemos de un teclado y con él podemos dar órdenes e instrucciones, obteniendo una respuesta a través de la pantalla de un televisor. Desde luego, este diálogo es apasionante. Sin embargo, el ordenador sólo comprenderá el lenguaje que tiene establecido, es decir, el Basic.

Si confiamos en nuestra máquina, ésta nos señalará los errores de sintaxis que podamos cometer al introducir las instrucciones. Las principales se desarrollan dentro de este libro y todas se repiten en un anexo, acompañadas de comentarios y ejemplos ilustrados. También se incluyen importantes notas para la utilización de las variables del sistema, tales como la redefinición de uno o varios nuevos juegos de caracteres, la extensión de los gráficos definidos por el usuario en más de 21 caracteres, etc...

A partir de aquí, será fácil profundizar sus conocimientos y escribir sus propios programas de gestión, enseñanza, juegos y demás.

Un ibro de gran utilidad plagado de numerosos programas ejemplo.

LIBROS



PROGRAMAR EN BASIC

EDITORIAL MARCOMBO.
Gabriel Domínguez
203 páginas.

Todas las explicaciones contenidas en este libro son de aplicación para dos casos especiales: tanto si el ordenador dispone de minidiscos como si se trabaja en cassette.

Consecuentemente, aunque se disponga de minidisco para el ordenador, el estudio de este volumen le será indispensable, ya que puede considerarse como los cimientos de la Programación. Además, casi todos los ordenadores provistos de minidiscos tienen una puerta de entrada/salida para el cassette, y todavía hay en el mercado algunos millares de programas en este medio, fácilmente traspasables al minidisco.

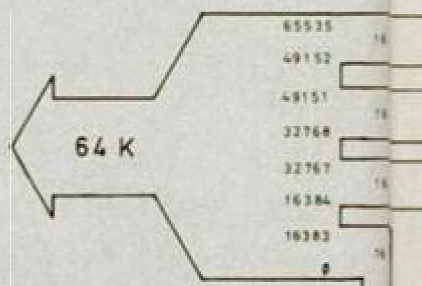
A lo largo de este libro se explicará, del modo más simple y concienzudo posible, las herramientas del lenguaje Basic, es decir, para qué y qué son, las palabras Reservadas. Con éstas se podrán construir excelentes programas, siempre y cuando se disponga de la suficiente imaginación.

Después de la propia explicación que nos ofrece esta obra, se añaden uno o más ejemplos para la fácil comprensión de los mismos, aunque éstos no siguen un orden alfabético, sino otro que podríamos llamar de utilidad. También se añaden todos aquellos trucos o rutinas que nunca aparecen en los manuales y significan grandes ayudas para el futuro programador.

LA MEMORIA PAGINADA DEL 128K

Jesús ALONSO RODRÍGUEZ

En este artículo, primero de una serie donde iremos haciendo un acercamiento progresivo al nuevo Spectrum Plus 2, empezaremos a ver la curiosa forma en que tiene organizada la memoria este ordenador.



Una de las características principales del Spectrum Plus 2 (no la única), es la de disponer de 128 K de memoria RAM. Ahora bien, estos 128 Ks de RAM no son directamente accesibles al programador —ahora veremos porqué— aunque esto no va a representar ningún problema, ya que vamos a ser capaces de utilizarlos por completo. Para ello, es importante comprender como está organizada la memoria del ordenador y familiarizarnos con algunas técnicas de programación bastante utilizadas en máquinas de superior categoría.

Partiremos de la base de que este artículo puede estar siendo leído por un usuario que acaba de adquirir su ordenador y carece por completo de conocimientos informáticos. Si no es su caso, tal vez pueda saltarse unas cuantas líneas, aunque es probable que no le venga mal refrescar algunos conocimientos.

ROM y RAM

Lo primero que se pregunta el neófito es qué quiere decir eso de que el Spectrum Plus 2 tiene 32 Ks de ROM y 128 Ks de RAM. Veamos primero qué es la ROM.

La palabra ROM está compuesta por las siglas «Read Only Memory» (Memoria de Solo Lectura). Se trata de un dispositivo electrónico que contiene una información que sólo puede ser leída, pero no alterada. Podemos compararlo con un libro.

La palabra RAM significa «Random Access Memory» (Memoria de Acceso Aleatorio). Se trata de un dispositivo electrónico capaz de almacenar información (luego veremos como) que puede ser escrita y leída. Podríamos asimilarlo a un cuaderno.

De la misma forma que un libro o un cuaderno están distribuidos en páginas y cada página en líneas, la memoria (sea ROM o RAM) está distribuida en lo que denominamos «posiciones de memoria». Cada posición de memoria es una «cajita» donde podemos almacenar un número entero positivo comprendido entre «0» y «255» ambos inclusive. Podemos decir que tal posición de memoria contiene un «0», tal otra un «200» o tal otra un «7». De momento, lo importante es que recordemos que una posición de memoria sólo puede contener un número comprendido entre «0» y «255», por tanto, todo lo que almacenemos en el ordenador será reducido, de alguna forma, a números. Llamaremos «Byte» a un número entero positivo comprendido entre «0» y «255», es decir, a un número que cabe en una posición de memoria, por tanto, podemos decir que cada posición de memoria almacena un byte.

Por otro lado, cada posición de memoria está especificada por un número, es decir, tiene una dirección. Es algo similar a los apartados de correos. Así, podemos afirmar que la posición de memoria número 357 contiene el byte «22» (por ejemplo), o que la posición 37548 contiene un «3». Las posiciones de memoria están numeradas desde «0» hasta «65535». Aunque, en realidad, el ordenador tiene más de 65536 posiciones de memoria, sólo puede acceder directamente a éstas, y de ahí viene todo el problema de la paginación.

CPU

La memoria (ROM y RAM) es uno de los componentes básicos del ordenador, pero no el único. Existe un componente más importante aún al que podríamos denominar el «corazón» (¿o sería más apropiado el «cerebro»?) del ordenador. Este componente es el «microprocesador», al que para abreviar llamaremos en lo sucesivo «la CPU». La palabra «CPU» está formada por las iniciales de «Central Procesing Unit» (Unidad Central de Proceso). En el Spectrum Plus 2, la CPU está constituida por un circuito integrado («Chip») de 40 patillas fabricado por la empresa «Zilog» y cuyo código es «Z-80A».

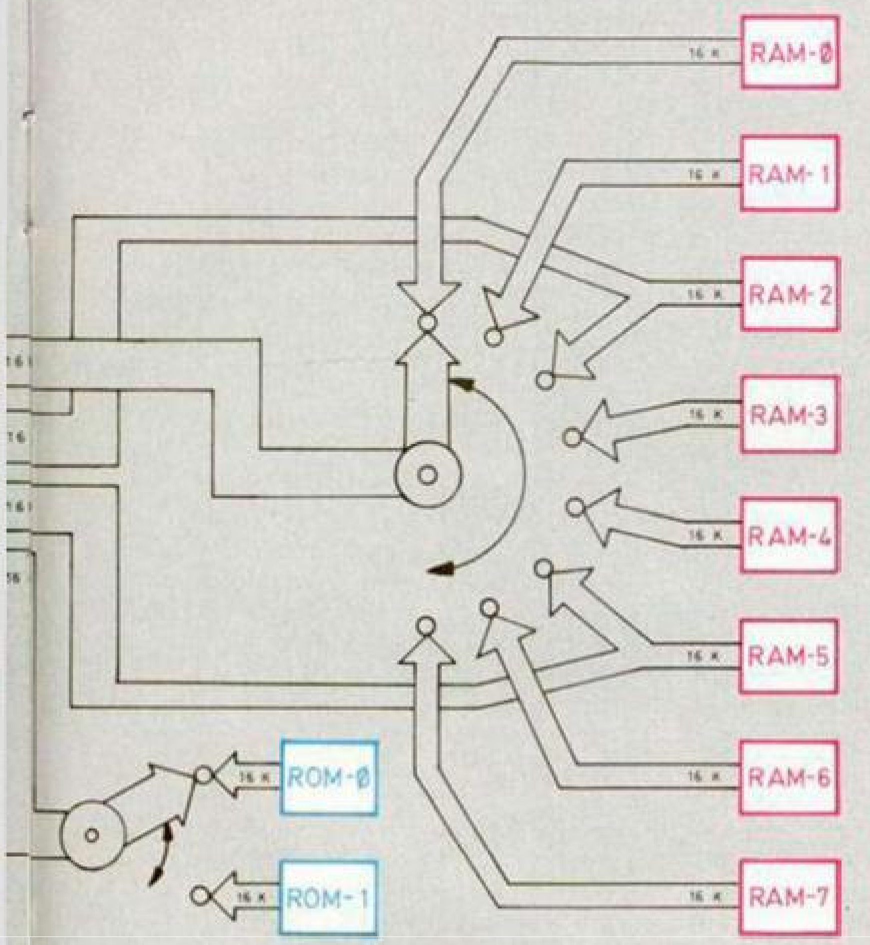
El Spectrum Plus 2, al igual que casi todos los ordenadores, es una «máquina de arquitectura Von Newman». Esto quiere decir que dispone de una memoria donde se almacenan tanto programas como datos, que tiene una CPU que ejecuta los programas, y que los procesos se realizan de forma secuencial.

Por lo tanto, la CPU tiene que comunicarse con la memoria. Para ello utiliza dos caminos: el bus de direcciones y el bus de datos. Cada vez que quiere leer una posición de memoria, escribe su dirección en el bus de direcciones y recibe su contenido a través del bus de datos. Si lo que desea es escribir en una determinada posición de memoria, escribe su dirección en el bus de direcciones y el dato que quiere guardar, en el bus de datos.

Como el contenido de una posición de memoria tien que ser un byte (número entero positivo comprendido entre «0» y «255»), por el bus de datos sólo pueden circular números de un byte. Sin embargo, el bus de direcciones es más «ancho», así que por él pueden circular números enteros positivos comprendidos entre «0» y «65535». A estos números los denominados «números de dos bytes».

Paginación

De todo esto se deduce que la CPU no puede trabajar, simultáneamente, con más



de 65536 posiciones de memoria (desde la «Ø» hasta la «65535»). Sin embargo, el Spectrum Plus 2 tiene 16384Ø posiciones de memoria, es decir, dos veces y media más. De ellas, 32768 corresponden a memoria ROM y 131Ø72 a memoria RAM. Parece un desperdicio tener dos veces y media más memoria de la que se puede utilizar, así que lo fabricantes de ordenadores han ideado un ingenioso sistema para poderla emplear toda. La idea es dividir la memoria en bloques de 16384 posiciones. A cada uno de estos bloques lo denominaremos «banco de memoria». El Spectrum Plus 2 tiene 10 bancos de memoria, de los cuales 2 son de ROM y 8 son de RAM. En cada momento, sólo puede trabajar con 4 de estos bancos que serán siempre 1 de ROM y tres de RAM. Podemas imaginar el bus de direcciones dividido en cuatro «carriles» a los que llamaremos «páginas». Cada página corresponde a un banco. La primera página será, siempre, de ROM y las otros tres de RAM. Podemos conectar un banco a cada página, y de esta forma, cada vez habrá cuatro bancos conectados a la CPU, pero podremos acceder a cualquier otro por el simple procedimiento de conectarlo a una de las páginas. Este sistema de conectar, alternativamente, uno u otro banco de memoria se denomina «paginación».

La primera página corresponde a las direcciones comprendidas entre «Ø» y «16383», la segunda, de «16384» a «32767», la tercera va de «32768» a «49151» y la cuarta de «49152» a «65535».

Los bancos de ROM se numeran como «0» y «1» y los de RAM de «0» a «7». A la primera página sólo podemos conectar el banco «ROM-Ø» ó el «ROM-1». El banco «RAM-5» está conectado permanentemente a la segunda página, el «RAM-2» a la tercera y, finalmente, a la cuarta página podemos conectarle cualquier banco de RAM, desde el «RAM-Ø» al «RAM-7».

En la FIGURA-1 se ha hecho un esquema muy simplificado de las posibles conexiones entre el bus de direcciones y los bancos de memoria. Los dos «conmutadores» representados por flechas giratorias sirven para visualizar la forma en que se conmutan los bancos de memoria. En la realidad, la elección de uno u otro banco se hace almacenando en un determinado registro, aunque, de momento, dejaremos que sea el Sistema Operativo quien se ocupe de conmutar los bancos de memoria.

Normalmente, el ordenador trabaja con ambos bancos de ROM en la primera página («ROM-Ø» para el editor y «ROM-1» para el intérprete) y con los bancos «RAM-Ø» y «RAM-7» en la cuarta página. El resto de los bancos, «RAM-1», «RAM-3», «RAM-4» y «RAM-6» se utilizan por el Basic como un «DISCO-RAM», es decir, como un sistema de almacenamiento que funciona como un disco, si bien, con la ventaja de un acceso más rápido y el inconveniente de que los datos se pierden al desconectar la alimentación.

En sucesivos artículos, iremos viendo la forma de utilizar el «DISCO-RAM» para almacenar programas y datos, extendiendo enormemente la capacidad del ordenador.

Los 16 dedos de un ordenador

Llegado a este punto, es muy probable que el lector se pregunte por qué todos estos números no son cifras «redondas». Sería más fácil tener 32.ØØØ posiciones de ROM y 128.ØØØ de RAM. Lo cierto es que estos números no son redondos para nosotros, pero sí lo son para el ordenador.

La humanidad empezó a contar con los dedos, y dio la casualidad de que tenía 1Ø, así que, cuando se les acababan los 1Ø dedos, tenían que empezar la cuenta de nuevo. Como resultado, se inventó un sistema de numeración con 1Ø dígitos distintos (del «Ø» al «9»). Los ordenadores sólo utilizan dos dígitos («Ø» y «1», pero los agrupan de 4 en 4 para formar 16 combinaciones distintas, a cada una de estas combinaciones se le hace corresponder un dígito determinado, por lo que existen 16 dígitos distintos: del «Ø» al «9» y de la «A» a la «F». A este tipo de numeración se la denomina «hexadecimal». Un número escrito en decimal puede ser covertido en hexadecimal. Todos los números que hemos visto hasta ahora, son «números redondos» si los convertimos a hexadecimal. (I)

La memoria de un ordenador se suele medir en «Ks», un «K» son 1.Ø24 posiciones de memoria, por eso decimos que el Spectrum Plus tiene 131.Ø72 posiciones de memoria (131.Ø72 = 128 × 1.Ø24). Por cierto, 1.Ø24 en hexadecimal es 4ØØ. En el Plus II, cada banco tiene 16Ks, por lo que tenemos 2 × 16-32Ks de ROM y 8 × 16Ks = 128Ks de RAM. En cada momento, sólo hay conectados a la CPU 16Ks de ROM y 16 × 3 = 48Ks de RAM. El editor utiliza «RAM-7» como almacenamiento temporal, así que nos quedan 64Ks para el «DISCO-RAM».

(I) El sistema de numeración decimal no es el mejor posible, por más que los humanos nos empeñemos en creer lo contrario. Si tomamos el número 1Ø y lo dividimos por 2, obtendremos un 5 que es número primo, es decir, no podemos seguir dividiendo, a menos que saquemos decimales, de hecho, con cada división aparece un decimal más. Sin embargo, si tomamos el número 1Ø en hexadecimal (16 en decimal) y lo dividimos por dos, obtendremos «8», si lo volvemos a dividir, obtendremos «4», a continuación «2» y finalmente «1». Hemos podido llegar a la unidad partiendo de la base sin obtener ningún número primo. Esta es una de las múltiples ventajas que tiene el sistema hexadecimal sobre el decimal. Si algún día tomamos contacto con una cultura ajena a nuestro planeta, seguramente se sorprenderán mucho de nuestro extraño sistema de numeración.

BACKUP PARA EL INTERFACE BETA

José Manuel LAZO

Este programa ha sido realizado con el fin de rellenar el vacío de información que los usuarios del Beta-Disc se encuentran a la hora de buscar utilidades para disco.

Con la compra del interface Beta-Disc se adjunta un disco con utilidades para empezar a usar el mismo rápidamente. Entre estas utilidades se encuentra un "Backapeador". Con él se consigue, teniendo un solo drive, realizar copias íntegras de todo el contenido de un disco cualquiera. El mismo se basa en ir volcando en zonas de RAM trozos de un disco *master* y luego volcar las mismas en el disco *virgen*. Hasta aquí está muy bien, pero resulta desesperante su utilización debido a la inmensa cantidad de veces que tendremos que sacar y meter los discos: concretamente 30 en el peor de los casos. Puestos a estudiar este detalle, nos dimos cuenta de que, teniendo como tiene el Spectrum 48 K de memoria RAM, el programa sólo utiliza 24 para *scratch* de los datos. Aparte de esto el *backup* antiguo tiene una longitud de 2.048 bytes, y el nuestro, que hace lo mismo, y además mucho mejor, tiene una longitud escasa de 190 bytes. Resulta inconcebible cómo es posible que una cosa que se pueda hacer con 190 bytes de memoria dejando el resto como buffer de RAM se tenga que hacer con un programa de 2 K's teniendo como buffer sólo 24 K.

Con el programa que os ofrecemos a continuación, esta operación de introducción y extracción de los discos, quedará reducida, en el peor de los casos, a un número de 18 operaciones. Merece la pena teclearlo.

Cómo funciona backup

Una vez cargado en memoria se nos pedirá que introduzcamos el disco máster en el drive, hecho lo cual podemos pulsar cualquier tecla para cargar el directorio del mismo. Nunca introducir un disco máster sin protegerlo previamente contra escritura y evitaréis para evitar desagradables sorpresas.

El directorio se carga en la memoria de pantalla, sobre esto es preciso decir que la misma se utiliza también como *scratch* del *backup* por lo que no es de estrañar que la misma se llene de basura. El programa backapeador, como ya habréis tenido ocasión de comprobar, está ubicado en la memoria de atributos.

El programa sólo "backupeará" las zonas del disco que contengan información, es decir, si el disco máster no estuviese lleno, sino que sólo tuviera ocupados 400 sectores. Por ejemplo, sólo se backapearán éstos haciendo la operación mucho más corta.

De ahí que lo que primero realiza *backup* es cargar el directorio en pantalla, para ver qué parte del disco está ocupada y cuál está libre. Una vez hecho esto entramos ya en la rutina backapeadora propiamente dicha. La misma espera, antes de realizar cualquier operación de carga o grabación, la pulsación de la tecla «Space» invitándonos a pulsarla con un pitido.

Esto es lo que tenemos que hacer la primera vez: pulsarla una vez que *backup* haya cargado el directorio y suene el pitido de aviso; eso sí, con el disco máster introducido en el drive.

A partir de ahora las operaciones son muy sencillas: cuando suene el pitido intercambiar los discos: máster por virgen o viceversa, y volver a pulsar «Space».

Cuando *backup* considere que el disco ya está copiando parará presentando el informe OK. Así de sencillo es el manejo: muy parecido, eso sí, el *backup* antiguo, pero mucho más racional que éste.

Pasemos ahora a ver cómo funciona este pequeño milagro, para lo cual, es preciso el fijarse en el listado 1 que contiene el programa fuente, a introducir con un ensamblador.

Lo primero que vemos es una rutina, DISCO, sita en la línea 100 y sucesivas. La misma actualiza unos registros con los valóres de la pista, sector, sitio en la RAM donde van a ir los sectores, número de sectores a cargar, y un status que indica grabar o cargar y llama a una importante rutina del DOS. Previamente a esta operación se ha paginado el mismo sobre la ROM y guarda en la pila la dirección de la rutina despaginadora, línea 100.

A continuación de la rutina de acceso directo al disco viene una tabla de valores, los mismos se utilizan después para saber en qué sitios de la memoria RAM van las 18 pistas que se pueden volcar de un tirón. Después de esta tabla está el programa propiamente dicho:

Desde la línea 360 a la 420 se carga el directorio en pantalla. En la línea 430 se coge en el registro A el puntero que se haya en el mismo que marca la última pista grabada del disco. A partir de aquí se calcula, con un sencillo algoritmo cuántos bloques de 18 pistas hemos de cargar para backapear del disco sólo la parte que contenga información.

Este número de veces está en el registro B en la línea 500 por lo que se hace un bucle desde esta línea hasta la 610 en la cual se carga una parte del disco máster y se graba en el virgen. Este bucle, después de actualizar las variables acordemente a lo que se pretende llama a la subrutina BLOQUE.

Aquí es preciso hacer un inciso: aunque se ha mencionado la palabra *variables* no son tales, dado que el programa para ocupar la menor cantidad posible de memoria funciona bajo la filosofía de modificarse a sí mismo según las operaciones que tenga que realizar.

La subrutina «bloque»

Esta subrutina carga en toda la memoria RAM disponible, incluyendo la pantalla, un bloque de 18 pistas del disco a partir de donde esté marcado el puntero en la última operación. Para ello se basa en la tabla que aludimos arriba.

Previamente a la carga o grabación, dado que se utiliza la misma subrutina para las dos operaciones, se espera la pulsación de la tecla «Space» a la vez que se realizan unos OUT's que tienen como misión el producir un pitido y unas rayas en el borde, como se pude de dedicir de la contemplación de las líneas 640 y sucesivas.

Caso de que haya que copiar el disco completamente, el último bloque no coincide con las pistas que queden en el disco por copiar. Esto se detecta coherentemente en la línea 830 y sucesivas, no produciéndose ningún error por ello.

Es menester decir en este momento, que la rutina del DOS que se utiliza chequea errores, pero si se produce alguno de los mismos no retorna al *backup*, sino que lo gestiona el DOS. Esto puede producir que si tenemos el disco virgen protegido contra la grabación el programa se pueda colgar, pero sin ninguna consecuencia posterior es decir, no perderemos ningún dato importante.

Instrucciones de tecleado

Para conseguir el *backup* hay dos caminos posibles: uno es, con un ensamblador,

introducir el listado fuente que acompaña al artículo. El otro es el siguiente:

— Primero teclea el listado Basic y sálvalo en una cinta con la orden: SAVE "BACKUP" LINE Ø.

— Luego, y con la ayuda del cargador universal de CM se introducen las datas del listado hexadecimal, y, después de hacer un DUMP en la dirección 4ØØØØ se salva el código objeto resultante en la misma cinta a continuacón del Basic con la orden SAVE del cargador, dando como nombre «backup», como comienzo la 4ØØØØ y como longitud 184 octetos.

A pártir de este momento ya lo podemos usar tecleando LOAD "" y cargando de la cinta donde la hayamos grabado. También es factible, y bastante aconsejable, el pasar este programa a disco, operación ésta muy sencilla.

DESENSAMBLE DE LA RUTINA

```
10 ; BACKUP PARA BETA-DISC
20 ; POR: J.M.LAZO
30 ; PARA: MICROHOBBY
40 ;
50          ORG     #5800
60          JP      START
70 ;
80 ; GO A CAPON AL DISCO
90 ;
100 DISCO   CALL    #3CAA
110         PUSH    HL
120 DIREC   LD      HL,0
130 SECPIS  LD      DE,0
140 STATUS  LD      A,0
150         LD      B,10
160         CALL    #B9F
170         RET
180 TABLA   DEFW    #4000
190         DEFW    #4A00
200         DEFW    #6000
210         DEFW    #6A00
220         DEFW    #7400
230         DEFW    #7E00
240         DEFW    #8800
250         DEFW    #9200
260         DEFW    #9C00
270         DEFW    #A600
280         DEFW    #B000
290         DEFW    #BA00
300         DEFW    #C400
310         DEFW    #CE00
320         DEFW    #D800
330         DEFW    #E200
340         DEFW    #EC00
350         DEFW    #F600
360 START   LD      HL,0
370         LD      (SECPIS+1),HL
380         LD      HL,16384
390         LD      (DIREC+1),HL
400         LD      A,0
410         LD      (STATUS+1),A
420         CALL    DISCO
430         LD      A,(#48E2)
440         LD      C,1
450 LOOP1   SUB     18
460         JR      C,YAESTA
470         INC     C
480         JR      LOOP1
490 YAESTA  LD      B,C
500 LOOP2   PUSH    BC
510         LD      A,0
520         LD      (STATUS+1),A
530         CALL    BLOQUE
540         LD      A,(SECPIS+2)
550         SUB     18
560         LD      (SECPIS+2),A
570         LD      A,255
580         LD      (STATUS+1),A
590         CALL    BLOQUE
600         POP     BC
610         DJNZ    LOOP2
620         RST     #8
630         DEFB    255
640 BLOQUE  LD      A,B
650         OUT     (#FE),A
660         DJNZ    BLOQUE
670         LD      A,127
680         IN      A,(#FE)
690         BIT     0,A
700         JR      NZ,BLOQUE
710         LD      A,0
720         OUT     (#FE),A
730         LD      B,18
740         LD      IX,TABLA
750 LOOP4   PUSH    BC
760         LD      L,(IX)
770         INC     IX
780         LD      H,(IX)
790         INC     IX
800         PUSH    IX
810         LD      (DIREC+1),HL
820         CALL    DISCO
830         LD      A,(SECPIS+2)
840         CP      #9F
850         JR      Z,YANO
860         INC     A
870         LD      (SECPIS+2),A
880         POP     IX
890         POP     BC
900         DJNZ    LOOP4
910         RET
920 YANO    POP     HL
930         POP     HL
940         LD      A,162
950         LD      (SECPIS+2),A
960         RET
970 ZINAL
```

```
Pase 2 errores:00

BLOQUE 5878    DIREC  5807
DISCO  5803    LOOP1  5852
LOOP2  585A    LOOP4  588F
SECPIS 580A    START  5839
STATUS 580D    TABLA  5815
YAESTA 5859    YANO   58B3
ZINAL  58BB

Tabla usada:  172  de   267
```

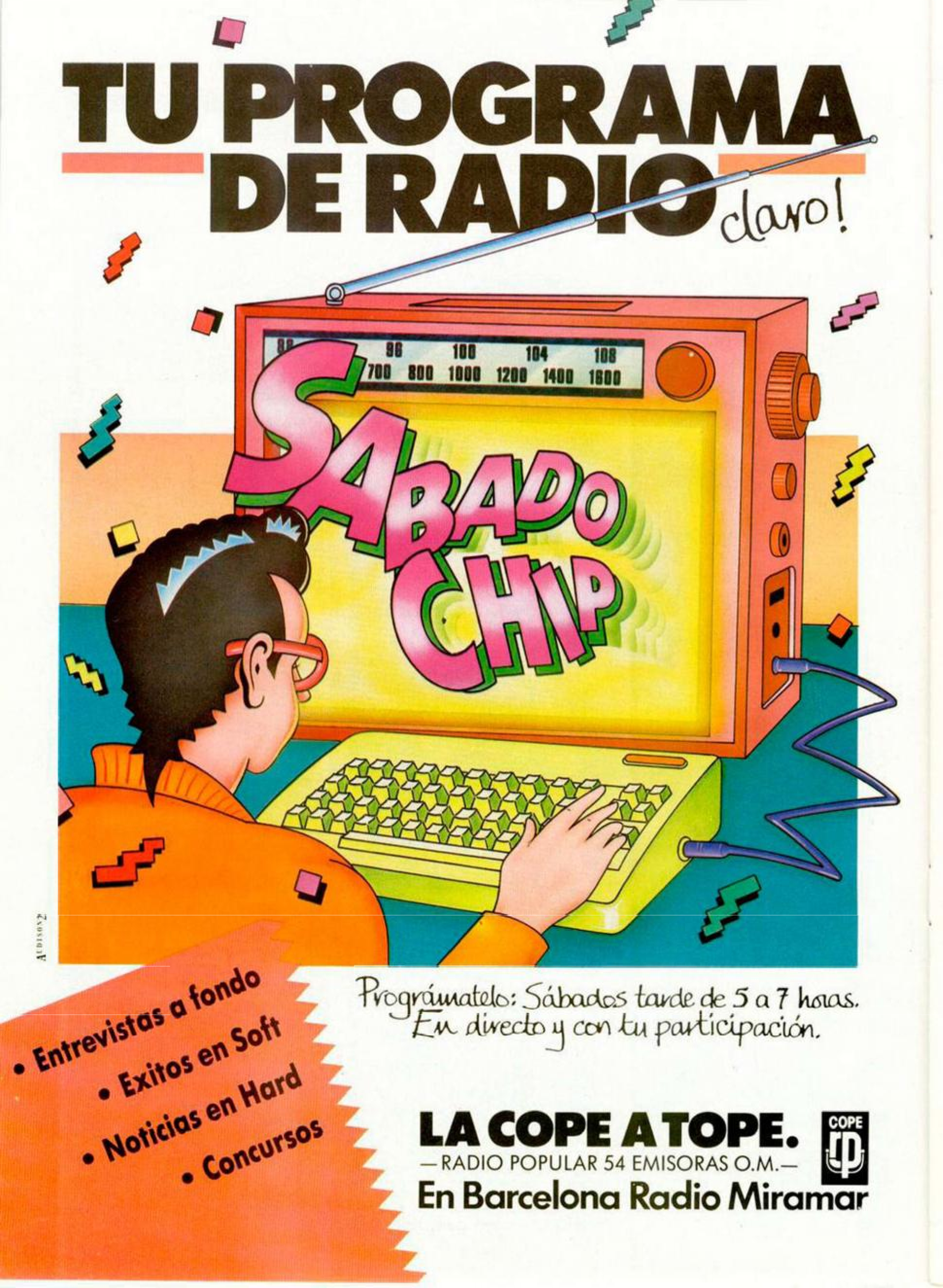
PROGRAMA BASIC

```
10 REM BACKUP BETA-DISC
20 REM POR J.M.LAZO
30 BORDER VAL "0": PAPER VAL "0": INK VAL "4": POKE VAL "23624",VAL "4": CLEAR VAL "24575"
40 RANDOMIZE USR VAL "15363": REM : LOAD "backup"CODE 22528,187
50 PRINT AT VAL "10",VAL "0";"INSERTA DISCO MASTER": PAUSE VAL "0": RANDOMIZE USR VAL "22528"
```

LISTADO CÓDIGO MÁQUINA

LINEA	DATOS	CONTROL
1	C33958CDAA3CE5210000	1037
2	1100003E00060ACD9F0B	470
3	C90040004A0060006A00	541
4	74007E00880092009C00	680
5	A600B000BA00C400CE00	930
6	D800E200EC00F6210000	957
7	220B5821004022085B3E	422
8	00320E58CD03583AE248	804
9	0E01D61238030C18F941	656
10	C53E00320E58CD78583A	882
11	0C58D612320C583EFF32	849
12	0E58CD7858C110E4CFFF	1414
13	78D3FE10FB3E7FDBFECB	1717
14	4720F33E00D3FE0612DD	1118
15	211558C5DD6E00DD23DD	1147
16	6600DD23DDE5220858CD	1143
17	03583A0C58FE9F280A3C	772
18	320C58DDE1C110DDC9E1	1452
19	E13EA2320C58C9000000	800

DUMP: 4.0000
N.° BYTES: 187


TU PROGRAMA
DE RADIO
claro!
96 100 104 108
700 800 1000 1200 1400 1600
SABADO CHIP
• Entrevistas a fondo
• Exitos en Soft
• Noticias en Hard
• Concursos
Prográmatelo: Sábados tarde de 5 a 7 horas.
En directo y con tu participación.
LA COPE A TOPE.
COPE
—RADIO POPULAR 54 EMISORAS O.M.—
En Barcelona Radio Miramar

TOKES & POKES

EQUINOX

A cerca de este maravilloso juego ya hemos hablado en más de una ocasión, pero como J. L. López Remacho (Torremolinos) nos quiere contar para qué sirven muchos de los objetos que aparecen en él, pues retomamos el tema.

Crédito: para viajar en el mismo nivel.

Dinamita: destruye paredes verticales de roca.

Pala: destruye suelos de roca.

Bomba: destruye a los enemigos de una pantalla.

Fusible: abre barrenas parpadeantes.

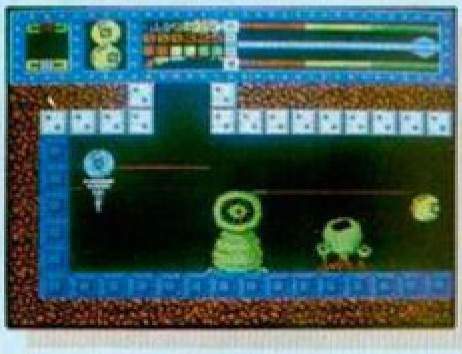
Batería Verde: repone municiones láser.

Bidón Rojo: repone energía.

Ticket numerado: para pasar de nivel.

Llave: abre puertas.

Frasco: se colocan en un horno rojo gigante.



HIJACK

POKE 63280, 201 tiempo infinito



SE LO CONTAMOS A...

• FCO. JAVIER GARCÍA

(Córdoba).

En el juego «**Spellbound**» de Mastertronic, se puede pasar a la pantalla de la bola con la botella que se encuentra en la pantalla anterior. Gracias a la «trompa del hombre fuerte» podrás invocar la presencia de cualquier personaje. Otra pista más para que no te quejes: si dejas caer la botella, aparece el Bawshee, que nos ayudará en el ascensor.

A continuación, envío un poke para que puedas resolver con mayor facilidad el «**Finders Keepers**»: POKE 34252,0, el cual te proporcionará vidas infinitas.

• DAVID CALANDRA REULA

(Santander).

Aquí tienes algunas soluciones para el problema que nos planteas de la aventura «**The Hulk**»; para poder salir del túnel debes pulsar el botón (Press Buttom) y una voz te contestará «Delay Om», con lo cual el gas que convierte en Bruce Banmer, retardará su salida. Con la fórmula habitual «Btt Lip» te conviertes en Hulk y así, la alta gravedad exterior no te matará. Para regresar de Field debes teclear Enter Dome seguida de la dirección West. De la pantalla «Small underground Room» podrás salir después de «Examine Room» y teclear «Watt» hasta que el veneno te ayude a salir. En el aérea «Fuzzy» debes dejar las gemas recogidas anteriormente, para que el ordenador, generosamente, te dé una ración extra de puntos.

• JOSÉ DOMINGO ROMERO TORIBIO

(Barcelona).

Después de rehusar en nuestros archivos hemos encontrado un par de pokes del **Dynamite Dan II:** POKE 32575,201 para inmunidad ante los enemigos. POKE 32587,201 para ir a otra pantalla, si se cae el agua.

ALIEN HIGHWAY

Miguel Viscarret, de San Sebastián, nos ofrece algún que otro poke de utilidad. Aquí va el primero:

POKE 43623,201 energía infinita.

POKE 39412,201 energía infinita.



LIGHTFORCE

Y ya que vamos de inmunidades, M. A. Tones Egea, catalán él, nos facilita un pequeño cargador para obtener esta considerable ventaja en este programa de ambiente aéreo.

```
10 BORDER 0: PAPER 0: INK 0
CLEAR 25000: LOAD "" CODE:
LOAD "" CODE: POKE 40673,9:
RANDOMIZE USR 18434.
```

PYRACURSE

Y aquí el otro. Éste sirve para hacerte inmune.

POKE 33450,201



CONSULTORIO

NÚMEROS ALEATORIOS

Quisiera saber cuál es la dirección décima de la ROM en la que se halla la rutina que genera números aleatorios y cómo (si es posible) poder usarla desde Código Máquina.

Carlos MEDIAVILLA-**Cantabria**

■ La rutina que genera números aleatorios funciona a través del calculador, por lo que no se puede usar directamente. Lo único que podemos hacer es remitirle al último capítulo de nuestro Curso de Código Máquina donde se explica el manejo del calculador, ya que es un tema demasiado complejo para abordarlo en el reducido espacio de que disponemos en esta sección.

No obstante, si lo que desea es generar números aleatorios en Código Máquina tal vez no sea preciso que recurra a la rutina de la ROM. Existen otras formas más sencillas de hacerlo. Le vamos a recomendar una que genera unos números bastante aleatorios comprendidos entre "0" y "9", es decir, de un byte. El procedimiento consiste en leer el contenido del octeto inferior de la variable del Sistema "FRAMES" (dirección 23672) y hacerle un XOR con el contenido del registro "R". El resultado es suficientemente aleatorio para la mayor parte de las aplicaciones.

PROGRAMAS

Cuando se da el caso de copiar un programa Basic y junto a él un programa en C/M, ¿cómo se puede introducir el programa en C/M sin que se borre el Basic?

En segundo lugar, quisiera saber qué significado tiene un programa en C/M a continuación de un programa Basic, ¿esto se hace por capricho o por necesidad?

En tercer lugar me gustaría hacerles la siguiente sugerencia: se trata de saber por qué en la sección de programas de lectores no publican ustedes más programas de utilidad.

Isidoro RAMOS-**Sevilla**

■ La respuesta a su primera pregunta es tan sencilla como evidente: introduzca primero el programa en Basic, sálvelo en cassette, introduzca después el Código Máquina y sálvelo a continuación del bloque Basic. Así de sencillo.

La razón de que se utilicen ambos lenguajes es que hay cosas que se hacen mejor en Basic y otras que sólo se pueden hacer en C/M, de forma que se utiliza un programa en Basic con llamadas a rutinas en C/M.

Los programas de utilidad solemos reservarlos para nuestra sección «Utilidades».

SOBRE EL PLUS II

Quisiera saber si en el Plus II con algún lápiz óptico, se pueden realizar mezclas de colores. En caso afirmativo, ¿qué lápiz óptico necesitaría?

Iñigo RUIZ-**Vizcaya**

■ Para hacer mezclas de colores, tanto en el Spectrum Plus como en el 48 K, no es necesario ningún lápiz óptico, ya que se puede hacer de forma muy sencilla y en Basic. El procedimiento es generar un UDG en forma de tablero de ajedrez e imprimirlo con un color de tinta y otro de papel. Como resultado, obtendremos un cuadrado de un color que será la mezcla de ambos. Por si no tiene muy claro cómo definir el UDG en forma de tablero de ajedrez, ahí va la rutina:

```
FOR I=0 TO 6 STEP 2
POKE USR "A"+I,170
POKE USR "A"+I,1,85
NEXT I
```

Supongamos que los colores a mezclar están en las variables «a» y «b». La forma de imprimirlo sería:

```
PRINT PAPER a; INK b; CHR$ 144
```

El joystick cuya construcción se publicó en nuestra revista, utiliza la norma «Kempston», por lo que resulta compatible con cualquier modelo de Spectru, incluido en el Plus II.

RED "NO TAN LOCAL"

Tengo dos ordenadores Spectrum y dos Interfaces 1, y quisiera conectarlos entre sí a una distancia aproximada de 300 metros aprovechando la red local de comunicación de los Interfaces 1. El problema es que vivo en la ciudad y lo que no puedo hacer es tirar un cable de 300 metros por medio de la calle. En consecuencia, mi pregunta es: ¿se pueden conectar dos Spectrum en red local, aprovechando la línea telefónica, el polo neutro de la red de 220 V, o dos emisoras que trabajen en la banda de 144 MHz?, ¿existe otra forma de hacer esto?

Matías CASAS-**Barcelona**

■ En su caso, más que de una «red local», se trataría de una «red no tan local», ya que 300 metros se encuentra fuera de todas las previsiones del fabricante en cuanto al alcance de su red. No podría hacerlo ni siquiera por cable, ya que las pérdidas de éste impedirían la comunicación.

Por supuesto, existe una forma de hacerlo que es la correcta. Se trata de utilizar dos «Modems» conectados a la salida RS-232 y enlazados entre sí, bien por teléfono, bien mediante dos emisoras.

Probablemente esto no le solucione el problema, porque no esté dipuesto a comprarse dos modems. Vamos a intentar una solución más sencilla.

Dado que tiene la posibilidad de utilizar dos emisoras de 144 MHz, y que éstas dan bastante calidad de modulación, puede intentar conectar la salida MIC de cada ordenador con la entrada de micrófono de cada emisora, y la entrada EAR con la salida de auriculares. De esta forma, podrá transmitir y recibir sin necesidad de utilizar, ni siquiera, el Interface 1. No le garantizamos que funcione, pero hay quien se comunica de esta forma incluso en 27 MHz, por lo que creemos que, con un poco de suerte, en 144 funcionará mucho mejor.

Por cierto, si le funciona, ¿qué le parece la idea de utilizar dos «controladores domésticos» para conmutar las emisoras y hacer la comunicación totalmente automática? Esperamos que nos escriba contándonos sus resultados.

... Y ADEMÁS, ES IMPOSIBLE

¿Existe en el mercado alguna EPROM que, de conexión externa, rompa la compatibilidad del Spectrum y le dé la posibilidad de colocar más colores en cada carácter?

José L. CORREA-**Madrid**

■ Dicen que «lo que no puede ser, no puede ser..., y, además, es imposible». Esto se aplica particularmente a lo que usted pide, ya que ni existe el dispositivo, ni hay posibilidad de que llegue a existir. El número de colores por carácter que tiene el Spectrum, viene definido por la propia arquitectura del ordenador, no sólo por la ROM. Lo que hace esta última es adaptarse a lo que le impone dicha arquitectura. Para variar el número de colores, habría que cambiar, no sólo la ROM, sino fundamentalmente la ULA, ya que es la exploración que ésta hace del archivo de pantalla, lo que define la resolución gráfica del ordenador. Si al Spectrum le quitamos la ULA y la ROM, lo que nos queda es un microprocesador y una memoria RAM, elementos ambos que son comunes a cualquier ordenador. Por tanto, cambiar la ULA y la ROM equivaldría a construir un ordenador totalmente distinto que nada tendría que ver con el Spectrum.

No obstante, hay un procedimiento de conseguir varios colores utilizando una rutina en Código Máquina sincronizada con la interrupción, que cambie los datos del archivo de atributos en medio de la exploración. El sistema permite conseguir efectos curiosos de varios colores por carácter, pero no resulta práctico, ya que sólo es posible aplicarlo sobre una determinada línea de pantalla y enlentece mucho la ejecución de los programas.

La conclusión que podemos sacar es que no existe un método que, de forma general, cambie la resolución gráfica del ordenador, ni es previsible que llegue a existir. Nosotros, a veces, bromeamos sobre ello y le llamamos «la rutina del medio pixel». No obstante, si alguien lo descubre, ¡¡por favor, que nos lo cuente!!

LA MEMORIA DEL PLUS II

He adquirido un Spectrum Plus II y aunque domino el Basic a la perfección y el CM del Z-80 (perdón por la inmodestia), considero totalmente insuficiente la información acerca de las enormes posibilidades que ofrece el tener una cantidad de memoria mucho mayor que en un Spectrum normal, que era el que tenía anteriormente.

En primer lugar, desearía saber cómo poder utilizar la memoria adicional (aparte de las ventajas del Disco-Ram que ésas sí las conozco, pero en él sólo caben 64 K con lo que no ocupo la totalidad de la memoria. 64+48-112 K). Me explico: quisiera que me dijérais cómo trabajar con la memoria hasta llenar los 128 K, cómo grabar el resultar y cómo poder cargarlo, pues

por más que leo y releo el manual, no me aclaro. Según dice, el port 32765 es el encargado de gestionar la memoria adicional, pero no dice cómo se gestiona dicha memoria (o al menos, no lo suficientemente claro), es más, previene de que su uso provocará casi siempre la caída del sistema, que lamentablemente suele ser así, por cierto.

¿Cómo se direcciona qué ROM deseo? Pues al tener 32 K hay rutinas que podrán ser útiles y aún no sabemos utilizarlas.

José A. MOLINA-**Cádiz**

■ Efectivamente, el manual es bastante parco en información, por lo que no hay más remedio que «leer entre líneas» y acudir a otras fuentes. En principio, podemos decirle que por el port 32765 accede a un registro que controla la ROM que se está utilizando, el bloque de RAM que se encuentra en los 16 K superiores, el bloque que se utiliza de pantalla y el modo en que está trabajando el ordenador. Todo esto depende de los bits almacenados en ese registro. Podemos escribir el registro pero, por desgracia, no podemos leerlo. Todo esto lo pone en el manual, y también pone que, si se escribe algo en este port, lo más probable es que el ordenador se «cuelgue». Lo que no dice es por qué se «cuelga», pero esto se lo vamos a decir nosotros, ya que es importante saberlo para conocer cómo funciona la memoria.

Empecemos con la ROM. Existen dos bloques, ROM-Ø y ROM-1. Cuando se enciende el ordenador, queda activa la ROM-Ø que contiene las rutinas de edición utilizadas en modo 128. Cuando se ejecuta el comando «RUN» (o «GO TO») se activa la ROM-1 que es la que contiene las rutinas encargadas de interpretar los programas, es decir, el intérprete de Basic. Por tanto, si hacemos un «OUT» a ese port, en ese momento tendremos activa la ROM-1 ya que estamos ejecutando. Si nuestro «OUT» activa la otra ROM, el ordenador se pierde y sobreviene el «cuelgue». Así que recuerde, desde una rutina en CM, hay que retornar siempre teniendo activa la ROM-1.

Vamos con la RAM. El «Disco-RAM» sólo utiliza 64 K porque RAM-2 y RAM-5 están permanentemente paginadas en el segundo y tercer bloques. RAM-Ø está normalmente en el cuarto bloque y RAM-7 es una memoria temporal del editor, así que no nos quedan más que RAM-1, RAM-3, RAM-4 y RAM-6. Podemos paginar cualquier banco de RAM en los 16 K superiores, pero no olvide que la pila de máquina está aquí, así que sería conveniente cambiarla de sitio si no queremos que un «RET» se convierta en un desastre (ojo a las interrupciones, que normalmente solemos olvidar, pero también utilizan la pila). Por otro lado, cuando se esté ejecutando el Basic, tiene que estar activa RAM-Ø en los 16 K superiores, así que sólo podemos cambiar los bloques de RAM desde Código Máquina, pero no olvide que es muy importante tener activa RAM-Ø antes de intentar retornar a Basic.

Por último, recuerde que RAM-4 a RAM-7 están compartidas con la ULA que tiene prioridad de acceso en estos bancos; por tanto, las rutinas que requieran una temporización precisa deben corren en RAM-Ø a RAM-3.

En conclusión: si va a trabajar con programas mixtos en Basic y CM, olvídese de conmutar la RAM, deje que el Sistema Operativo se encargue de ello y utilice el «Disco-RAM» para guardar rutinas que funcionen en «overlay». Es más práctico y tendrá menos problemas. No se preocupe por perder 16 K de memoria, tampoco es para tanto.

Por el contrario, si su programa va a correr todo él en Código Máquina, puede conmutar los bancos libremente, pero deberá tener mucho cuidado con la pila y con las condiciones en que esté la memoria cada vez que llame a una rutina de la ROM.

Por nuestra parte, estamos recopilando toda la información que podemos sobre el 128 K. Ya tenemos preparados algunos artículos tanto de hardware como de software que iremos publicando en semanas sucesivas. Unos irán dedicados a quienes se han comprado éste como su primer ordenador y otros a quienes ya tienen cierta experiencia o han llegado al Plus II desde el 48 K (su caso). Pero no se preocupe, que no nos vamos a olvidar de este ordenador.

AVERÍA EN ISSUE 3B

Debido a una avería en mi Spectrum modelo ISSUE 3B, encontré quemada la resistencia R55. ¿Qué valor óhmico tiene?, ¿a qué parte del circuito pertenece?, ¿qué otros elementos pueden encontrarse mal o dónde puede estar el resto de la avería?

Luis PATIÑO-**C. Real**

■ El valor de la resistencia R55 es de 56 ohmios, 1/4 de vatio con una tolerancia del 5 por 100. Forma parte del circuito rectificador encargado de obtener —5 V a partir de los —12 V que genera el oscilador. No es fácil adelantar qué componentes pueden haberse averiado también, para ello, sería necesario examinar el circuito detenidamente y realizar una serie de medidas. No obstante, compruebe el diodo D-16 que es un zener de 5 V1 (5.1 voltios), así como el condensador C-47 que es un electrolítico de 22 microfaradios a 10 V, ya que se trata de los más posibles candidatos a causantes de que se haya quemado la resistencia.

"RTTY"

Si es posible, quisiera que me contestaráis a unas dudas que tengo sobre un programa publicado en vuestra revista número 92, página 26, referente a un receptor de RTTY. He tecleado el programa y funciona perfectamente en las frecuencias destinadas a este tipo de transmisiones, tanto comerciales como de aficionados, pero mis dudas son las siguientes:

La línea 345, ¿para qué sirve? puesto que el programa nunca llega a ella.

¿Cómo se podría hacer para introducir una opción que borrase la memoria de recepción?

¿Qué habría que hacer para que la información introducida en la memoria de recepción fuese pasada a una impresora?

Emilio CARLES-**El Ferrol**

■ El programa que nos indica fue remitido por un lector. Al no ser elaborado por nosotros, no estamos plenamente seguros de cómo funciona, pero lo hemos revisado detenidamente y hemos llegado a las siguientes conclusiones:

Las líneas 345, 35Ø, 355 y 36Ø no parecen tener ninguna utilidad, probablemente se trate de restos de versiones anteriores que el programador olvidó eliminar.

Para borrar la memoria de recepción, podrá valerle la siguiente rutina:

```
FOR n=338ØØ TO 65ØØØ
IF PEEK n=Ø THEN LET n=65ØØØ
POKE n,Ø
NEXT n
```

Ya que los datos se almacenan a partir de la dirección 338ØØ y una posición que contenga un «Ø» indica el final de los datos almacenados.

Para pasar el texto a impresora, puede utilizar una rutina muy similar:

```
FOR n=338ØØ TO 65ØØØ
LET dato=PEEK n
IF dato=Ø THEN LET n=65ØØØ
IF dato+31 THEN LPRINT CHR$ dato;
NEXT n
```

Esperamos que nuestro amigo Francisco Enguix nos perdone por modificar su programa.

LA SALIDA "REM"

¿Para qué sirve la salida «REM» de algunos cassettes?

Sergio L. SÁNCHEZ-**Huelva**

■ La indicación «REM» es una abreviatura de «REMOTE». Se trata de una toma de control remoto para el motor del cassette. Consiste en un jack de 2,5 mm con desconexión. Al introducirle una clavija, los dos terminales de ésta quedan en serie con la alimentación del motor, de forma que, si colocamos un interruptor entre ellos, el motor funcionará cuando el interruptor esté cerrado y se parará cuando esté abierto.

Esta conexión se coloca, siempre, a una distancia de 10 mm de la entrada «MIC», ya que hay micrófonos que incorporan un interruptor y utilizan una clavija doble en una sola pieza para su conexión.

CÓDIGO MÁQUINA

Últimamente en algunos programas publicados por MICROHOBBY, aparecen programas con líneas como éstas:

1 CF31D9E5D93EØ2CDØ116 1211
2 21ØØØØ22F6FC22F8FC22 1133

Me gustaría que me indicaran su significado.

Rafa BIEDMA-**Alicante**

■ Se trata de líneas de Código Máquina en el formato adecuado para ser introducidas mediante el Cargador Universal de Código Máquina MICROHOBBY publicado en los números 31 y 101 de nuestra revista.

Cada línea consta de tres partes: En primer lugar está el número de línea: los números de línea son siempre consecutivos. En segundo lugar hay 20 caracteres que, agrupados de 2 en 2, representan 10 bytes en hexadecimal. Por último, hay un número de control que es la suma de los 10 bytes de la línea.

La finalidad de utilizar este formato es minimizar la posibilidad de cometer errores al teclar un listado en Código Máquina.

OCASIONES

● **VENDO** por cambio de ordenador Zx Spectrum Plus, con todos sus accesorios: embalaje original, fuente, cables, etc., por el precio de 24.000 ptas. Regalo funda para el ordenador y un interface de joystick programable. Interesados llamar al tel. (986) 57 21 03. Pontevedra. Preguntar por Miguel.

● **VENDO** Spectrum 48 K con todos los cables, cassette para ordenador, por sólo 32.000 ptas. Además regalo numerosas revistas y funda protectora para cassette. Interesados escribir a la siguiente dirección: Alberto Escavi. Avda. Zamora, 1-5.° I. Vigo-3 (Pontevedra) o bien llamar al tel. (986) 41 79 41.

● **VENDO** interface, tipo Kempston marca DK'Tronics, por 1.000 ptas. Interesados llamar al tel. 407 35 36, de Madrid. Preguntar por Javi.

● **POR CAMBIO** de equipo vendo revistas y libros sobre el tema. Si te interesa pide información llamando al tel. (981) 78 29 52 (20 h), o bien escribe a la siguiente dirección: Manuel Cagigao. Apartado 2144. 15080 La Coruña.

● **VENDO** impresora Riteman F+compatible IBM comprado hace poco y con garantía, manuales en castellano e inglés, con maleta-transporte, 105 cps, con caracteres españoles, 8 K de buffer, 256 caracteres programables, itálicos y letra de alta calidad (NQL), por motivos económicos todo por 55.000 ptas., vendo también cable de conexión al ordenador. Interesados llamar al tel. 321 63 28, de Barcelona. Preguntar por Gema.

● **VENDO** Spectrum 128 K español, a estrenar. Incluyo lápiz óptico. Pon tú el precio. Preguntar por Andoni, en horas de oficina en el tel. 443 98 89, de Bilbao.

● **VENDO** Spectrum 48 K en perfecto estado, con su fuente de alimentación, cables, cinta de demostración y todo en su embalaje original, además interruptor Reset, con su joystick e interface Kempston, 2 libros de Basic y 15 revistas. Precio: 23.500 ptas. Pago gastos de envío. Interesados llamar al tel. (942) 27 35 59.

● **VENDO** Spectrum 48 K, junto con los libros, el transformador, cables de antena y de cassette, además de la cinta de demostración y el interface de joystick. Todo ello por sólo 20.000 ptas. Interesados escribir a la siguiente dirección: José Antonio Castillo Talledo. Gral. Mola, 23-7.° Izqda. Laredo (Cantabria) Tel. (942) 60 63 63.

● **VENDO** ordenador Zx Spectrum Plus, con más de 30 revistas del tema, 2 libros de programación, interface tipo Kempston, todo ello por sólo 45.000 ptas. Interesados escribir a la siguiente dirección: Miguel Angel García Cobos. Abad y la Sierra, 423-1.° Ibiza (Baleares).

● **VENDO** Spectrum 48 K, por 15.000 ptas. Incluye interface, joystick, etc. Interesados llamar al tel. (93) 890 42 04, de Barcelona. Preguntar por Jesús.

● **VENDO** polímetro marca ICE, modelo 680G, por sólo 45.000 ptas., con sus accesorios y en perfecto estado de funcionamiento. También vendo eco electrónico con modulador de frecuencia para un sinfín de efectos sonoros, por sólo 8.000 ptas. Interesados dirigirse a Pedro Sánchez García. Reina Victoria, 34-5.°B. Cartagena o bien llamar al tel. (968) 50 82 01.

● **URGE** vender ordenador Zx Spectrum Plus, con accesorios. Además regalo un joystick con su interface y una cinta de demostración. Precio: 28.000 ptas. Interesados escribir a Juan José Satue-Frías Rodríguez. Francisco Suárez, 29-3.° A. Valladolid.

● **VENDO** ordenador Zx Spectrum 48 K, fuente de alimentación, cables de carga, libro del Spectrum y cinta Horizontes, todo en perfecto estado, por sólo 17.000 ptas. Llamar al tel. (91) 457 92 99, de Madrid, y preguntar por Julio.

● **VENDO** Spectrum Plus, seminuevo (comprado en verano del 86), con cables, transformador, más de 25 revistas y un cassette Philips por un precio a convenir. Escribir a Fernando Almeida Parra. Zamora, 84-86, 3.° A. 37002 Salamanca. Llamar al tel. (923) 21 89 73 ó 21 85 85 (de 2 a 3 de la tarde).

● **VENDO** QL con fuente de alimentación, cables y libro de instrucciones, monitor color alta resolución. Precio total: 11.000 ptas. (negociables). Interesados llamar al tel. (925) 18 05 66. Tardes de 3 a 7. Preguntar por Pedro Luis. Santa Gema, 61. Quintanar (Toledo).

● **VENDO** Spectrum Plus, en perfecto estado, casi sin estrenar, con fuente de alimentación, cinta de demostración, manual de instrucciones e introducción al Basic, por sólo 30.000 ptas., junto cassette especial para ordenador Computone. Interesados llamar al tel. 89 61 83 o escribir a la siguiente dirección: J. J. Claudio. Reyes Magos, 14. San Fernando (11100 Cádiz).

● **URGE** vender Zx Spectrum 48 K. Adjunto cassette de ordenador, interface joystick. El precio es de 20.000 ptas. Enrique Naya. Tel. (91) 244 57 36, de Madrid.

● **VENDO** Spectrum 48 K, muy nuevo, con todos los accesorios, precio a convenir. Dirigirse a Toni Barros. Pza. San Eudaldo, s/n. Ripoll (Gerona). Tel. (972) 70 16 28 ó 70 28 76.

● **OFREZCO** Spectrum 48 K, manual, cinta de demostración, joystick, interface, con regalo de revistas. Magnífico precio. A ser posible sólo para los usuarios de Ciudad Real. Dirigirse a Ramón Camilo Alen Velázquez. Mata, 3, 2.° B. Ciudad Real. Tel. (926) 22 35 82.

● **VENDO** ordenador Zx Spectrum totalmente nuevo con salida para monitor, con todos los cables, cinta Horizontes, por un precio de 19.000 ptas. También vendo joystick Gran capitán por 1.000 ptas. junto o separado, el ordenador es de 48 K. Dirigirse a la siguiente dirección: Javier Piquer. Xaloc, 30. Plamós (Gerona). Tel. (972) 31 79 69. Llamar a partir de las 6 de la tarde.

● **VENDO** Spectrum 48 K, interface para joystick con amplificador de sonido. Además, regalo un cassette Computone (especial para ordenador), todo por 30.000 ptas. Dirigirse a Josele Gisbert Elio. Avda. Jacinto Benavente, 25. Pta. 7, 3.° esc. izqda. 46005 Valencia. O bien llamando al tel. 333 59 67 (en horas de comida o de 18 a 22 horas).

● **VENDO** Sikosha GP-50S, prácticamente sin usar y muy bien conservada, con un rollo de papel y cables, por sólo 20.000 ptas. Los gastos de envío corren de mi cuenta. Interesados escribir a la siguiente dirección: Manuel Losada Fernández. Tuy, 28, 5.°. Lugo.

● **VENDO** videojuegos Atari (consola, transformador, dos pares de mandos). Precio: 37.000 ptas. Interesados escribir a Alberto Lago. Paraguay, 27, 3.°. Vigo (Pontevedra).

● **VENDO** Zx Spectrum 48 K, en perfecto estado, con fuente de alimentación, cables, cinta de demostración y todos en su embalaje. Además, un interruptor Reset, joystick con interface Kempston, 2 libros y 15 revistas, todo por 23.500 ptas. Interesados llamar al tel. (942) 27 35 59. Preguntar por Sergio.

SINTONIZA CON MICROHOBBY Y LLEVATE GRATIS ESTOS FORMIDABLES RADIO-CASCOS

SUSCRIBETE AHORA Y LLEVATE LA MUSICA PUESTA

BENEFICIATE DE LAS VENTAJAS DE SER SUSCRIPTOR

- Un ahorro de más de 1.000 ptas.
- La comodidad de recibirla cada semana en tu domicilio.
- Evitar cualquier aumento de precio.
- **Llevarte, gratis, unos cascos con rádio incorporada (AM-FM).**

Oferta válida sólo para España, hasta el 31 de enero de 1987

Para suscribirte puedes llamar al (91) 734 65 00 o bien enviar tu solicitud a **Hobby Press, S. A.** Apartado de Correo 232. Alcobendas (Madrid).

GRATIS AL SUSCRIBIRTE



La maquina alucinante



Microprocesador Z80A. 128 K RAM. 32 K ROM. Teclado de 58 teclas. 32 columnas × 24 filas de texto. Gráficos de alta resolución (256 × 192 pixels). 8 colores con dos niveles de brillo cada uno. Calculadora en pantalla. 3 canales de sonido programables e independientes. Cassette incorporada. Salida TV y monitor RGB.

Interface MIDI (Musical Instrument Digital Interface). Salida Serie RS 232 bidireccional. Dos conectores para joysticks. Conector plano compatible con todos los modelos Spectrum anteriores. Editor de pantalla y dos versiones BASIC en ROM. 48 K BASIC, compatible con Spectrum 16 K, 48 K y ZX + . 128 K BASIC, compatible con ZX Spectrum 128.

Nuevo sinclair ZX Spectrum +2

C/. Aravaca, 22. 28040 Madrid. Tel. 459 30 01. Telex 47660 INSC E. Fax 459 22 92. Delegacion en Cataluña: C/. Tarragona, 110. Tel. 325 10 58. 08015 Barcelona.